1973-06

Simulation analysis of the TBM* memory system : an on-line mass storage system

Frazier, Robert Bruce

Monterey, California : Naval Postgraduate School

http://hdl.handle.net/10945/16799

SIMULATION ANALYSIS OF THE TBM* MEMORY SYSTEM: AN ON-LINE MASS STORAGE SYSTEM

Robert Bruce Frazier

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

SIMULATION ANALYSIS OF THE TBM* MEMORY SYSTEM: AN ON-LINE MASS STORAGE SYSTEM

by

Robert Bruce Frazier

Thesis Advisor: G. H. Syms

June 1973

Approved for public release; distribution unlimited.

Simulation Analysis of the TBM* Memory System:

An On-Line Mass Storage System

by

Robert Bruce Frazier
Ensign, United States Navy
B.S., Miami University, 1972

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
June 1973

ABSTRACT

The TBM* Memory System is an on-line, high density memory system with a capacity of up to three trillion bits. It is a cost efficient method of storing large data bases on-line with an effective response similar to disc storage.

Some features of the TBM* Memory System are: 1) Complete system redundance; 2) Modular expansion; 3) Hardware and data transfer reliability; 4) Large data base on-line; 5) Rapid response; and 6) Low cost per bit.

Major applications of a mass storage system are: 1) The replacement of large conventional on-line and off-line tape library systems; 2) Supplements to enlarge disc memory systems; and 3) the provision of on-line storage for large data bases.

Since the TBM* Memory System is indeed very complicated, a simulation model was developed to help the user understand the system and to predict the TBM* performance under various configurations and loading conditions. In order to help the user select the proper configuration for his installation, a sensitivity analysis is also provided.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENTS

The author wishes to thank his thesis advisor, Gordon Syms, for his help and patience during the entire research process; also to thank Eric Sabu and Wayne Bouker of the Ampex* Corporation for their help in explaining the TBM* Memory System.

I. INTRODUCTION AND OBJECTIVES

A. INTRODUCTION

More companies are converting their computer facilities to a centralized system with remote data entry locations. This trend tends to create very large data bases. Thus the demand for a quality mass storage system continues to increase. As companies continue to centralize, the number of installations with a mass storage system is definitely going to increase in the near future.

The TBM* Mass Storage System was selected as a research topic for several reasons:

1. The system is past the theoretical and development stages and into marketing and production stages.

2. Much more information, especially technical manuals and sales literature, was available.

3. The base location for the TBM* Memory System is a division of Ampex Corporation in Sunnyvale, California, which allowed many trips for interviews.

The Navy has many large data base applications, both as disc and tape libraries, which are reasonable candidates for a mass memory system. This research on performance characteristics of the TBM* Memory System should be of use to the Navy in the near future, and the knowledge and experience gained during the research should be of value in the writer's future shore assignments as a Naval officer.

B. OBJECTIVES

The TBM* Memory System apparently is a cost efficient method to store a large data base and still have the

effective access time of disc. Each installation has their own unique storage and performance characteristics.

The primary objectives are:

1. To give both a brief and an in-depth description of the TBM* Memory System.

2. To verify the hypothesis in the above paragraph.

3. To analyze performance characteristics of the TBM* System under varying system configurations, host loading factors, data arrangement criteria, etc.

4. To define applications where mass storage memory systems are practical, in particular, applications in which the TBM* Memory System could be used.

After reading the section entitled "The TBM* Memory System," the reader should understand the basic features and advantages of the TBM* System. After reading the section entitled "The TBM* Memory System Model," the reader should know how the TBM* Memory System operates and how the model represents the system. After reading the analyses, conclusions and recommendations, the reader should be able to infer what configuration of the TBM* Memory System would best suit his purposes. The appendices give the reader a more detailed understanding of various sections of the TBM* System, its uses, and other points of interest not included in the main body of the thesis.

II. THE TBM* MEMORY SYSTEM

The TBM* Memory System architecture and technology reflect a significant departure from conventional approaches to the problems of data storage and retrieval. Data is stored on magnetic tape, but this is where the similarity to conventional tape oriented devices ends. TBM* is a massive digital memory system with a maximum capacity equivalent to that of 30,000 computer tapes (800 bpi). TBM* is a SYSTEM; it is self-contained and capable of operating independently of other processors. It can be used in a stand-alone configuration or it can be interfaced to commercially available computers to act as an auxiliary memory system to the host computer.

TBM's unique system architecture provides such features as rapid random access to any record, extremely high data throughput (more than four times that of an IBM 2314 disc drive) and many more features, at a cost low enough to make the TBM the economical solution to massive data storage and retrieval problems.

TBM* utilizes standard two inch wide magnetic video tape of the type used in commercial television. Data is recorded in a block format with each individual block identified by a unique address. Rapid random access to any record is achieved by performing block address searches at tape speeds of 1000 ips - both forward and backward. Because of the high search speed and because of the high packing density, this corresponds

to searching through the equivalent of 10 conventional computer tapes per second. The system allows up to six simultaneous tape searches - the equivalent of searching 60 computer tapes per second.

Functionally, the system can be divided into two parts: the Data Storage Section, DSS, and the Communication and Control Section, CCS. The DSS consists of one or more of each of the three basic building blocks: Dual Transport Modules, transport drives and data channels. A system must have one each of the basic building blocks, while more units can be added to provide desired system capacity and throughput characteristics for each application. The CCS contains the TBM Supervisory Control Processor, SCP, External Data Channel Processors, EDCP, Staging discs and channel switches. The SCP is the control mechanism for the TBM system. It also handles host computer I/O requests, controls interface buffers, staging discs and switches, and monitors the overall system status.

A. ADVANTAGES OF THE TBM* MEMORY SYSTEM

1. Data Base Consolidation

TBM is the economical approach to consolidation of a data base into a single integrated and meaningful structure. It can store up to 400 billion bytes on line - the equivalent of 3.2 trillion bits. In order to judge the relative size of a trillion bits of data, the following comparisons are given. A trillion bits is equivalent to:

2,900	1600 bpi fully recorded 2400' tapes, or
3,500	2314 disc packs, or
1,000	3330 disc packs, or
950	Fastrand II drums, or
250	2321 data cells (400 Mbyte Units)

2. <u>File Maintenance</u>

TBM's approach to data handling enables users to access, process and update files faster and more frequently than with conventional techniques. With TBM's access methods and buffering techniques, only the desired records need be sent to the host computer for processing. (See Appendix E).

3. <u>Real Time Applications</u>

A multilevel priority processing structure, together with concurrent random access to multiple files make TBM an element of on-line real-time data communication system. (See Appendix F).

4. <u>Scheduling Flexibility</u>

TBM can facilitate job scheduling. There is no need to worry about retrieving and mounting tapes or disc packs - with TBM the data is available on-line.

5. <u>Data Security</u>

Protection against accidental overwriting is provided through read-only interlocks. File access protection is implemented via user codes associated with individual blocks or entire files. Backup data protection is provided by simultaneously writing data from a single EDCP through data channels to two topplates (tape mounting units). TBM tapes can also be copied independently of the host computer for further backup.

The TBM* Memory System provides an economical means of keeping several copies as well as several generations of files on-line. A single TBM tape is also an off-line storage unit containing the equivalent of <u>500</u> computer tapes at 800 bpi and fits in a space the size of a three inch binder.

6. <u>System Reliability</u>

High reliability is achieved through extensive component redundancy, multipath access and dynamic TBM device reconfiguration. See Appendix B for a more in depth explanation of data reliability and system reliability.

7. <u>Off-Line Operations</u>

System components may be placed in the off-line mode if disconnected from the host computer. The off-line mode is used for performing system diagnostics to locate problems, for preaddressing and testing tapes for routine maintenance, and for file duplication for backup protection. These operations are performed with a special off-line language and do not require the host CPU, the staging discs or the EDCP (External Data Channel Processor).

B. MODULARITY OF THE TBM SYSTEM

From the basic system, Figure 1, containing one each of the building blocks, the system can be expanded both in capacity and performance. Up to thirty-one Dual Transport Modules can be added to provide on-line storage equivalent to more than thirty thousand standard computer tapes. To provide for many simultaneous activities, up to six Transport Drivers

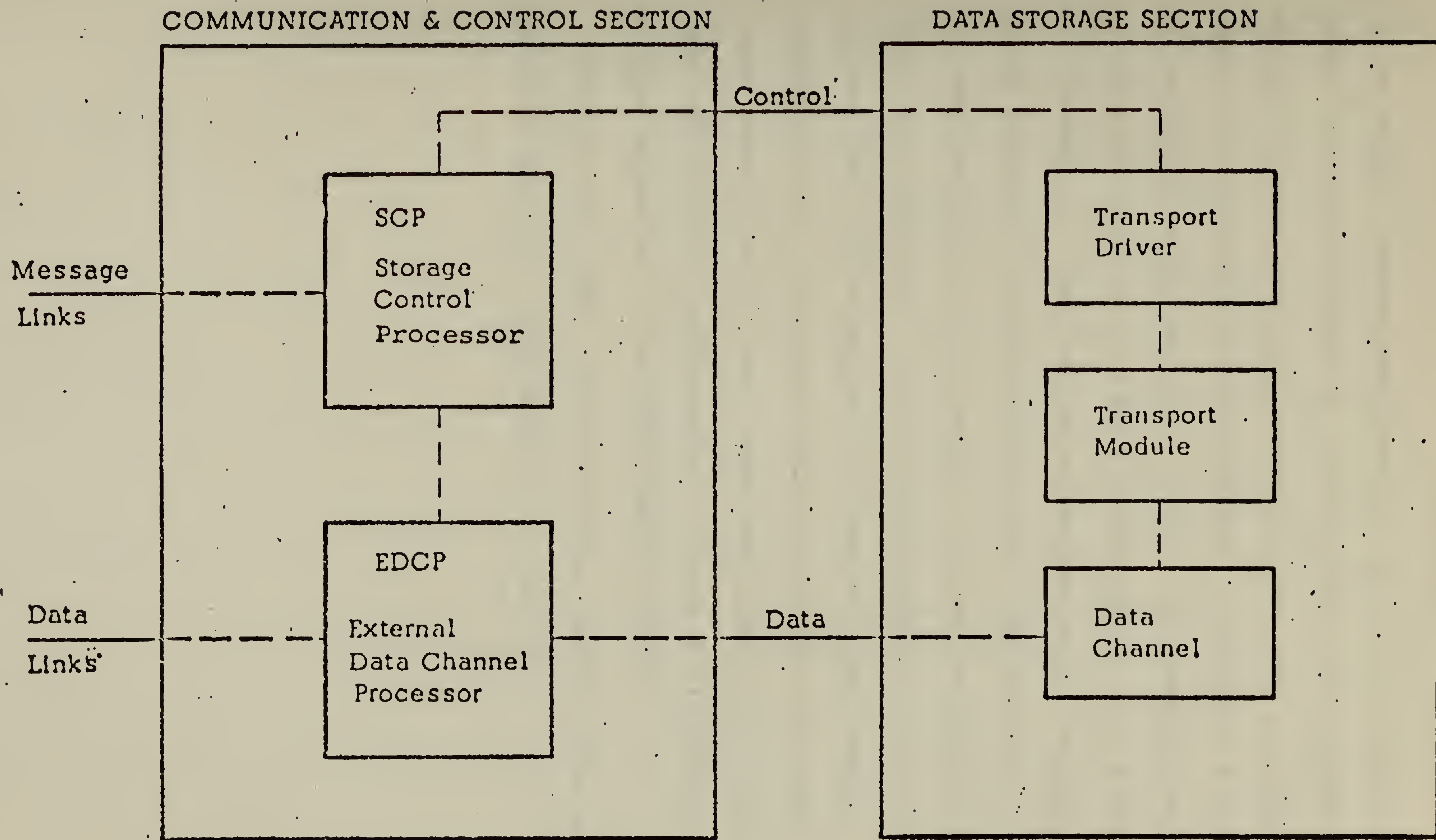


Figure 1 MINIMUM TBM* MEMORY SYSTEM CONFIGURATION

can be added. These make possible up to six concurrent random accesses (Figure 2). With added Data Channels and another External Data Channel Processor, two simultaneous read/write commands may be executed, yielding 1.5 megabytes per second data throughput. (Note: The original system proposed by Ampex was to have a throughput of 4.5 megabytes per second, but this was later limited to 1.5 megabytes due to the capacity of selected computer for the EDCP.) Switching matrices permit any Transport Driver and Data Channel to access any topplate. As the system expands, so do the switching matrices. This approach allows multipath access, dynamic reconfiguration, and optimum device scheduling, which assures a continuous flow of data, even when part of the system is taken off-line for maintenance. Figure 2 illustrates the maximum expansion of the TBM* Memory System.

Much of the above information was derived from Ampex's technical manuals, References 1, 2, and 3.

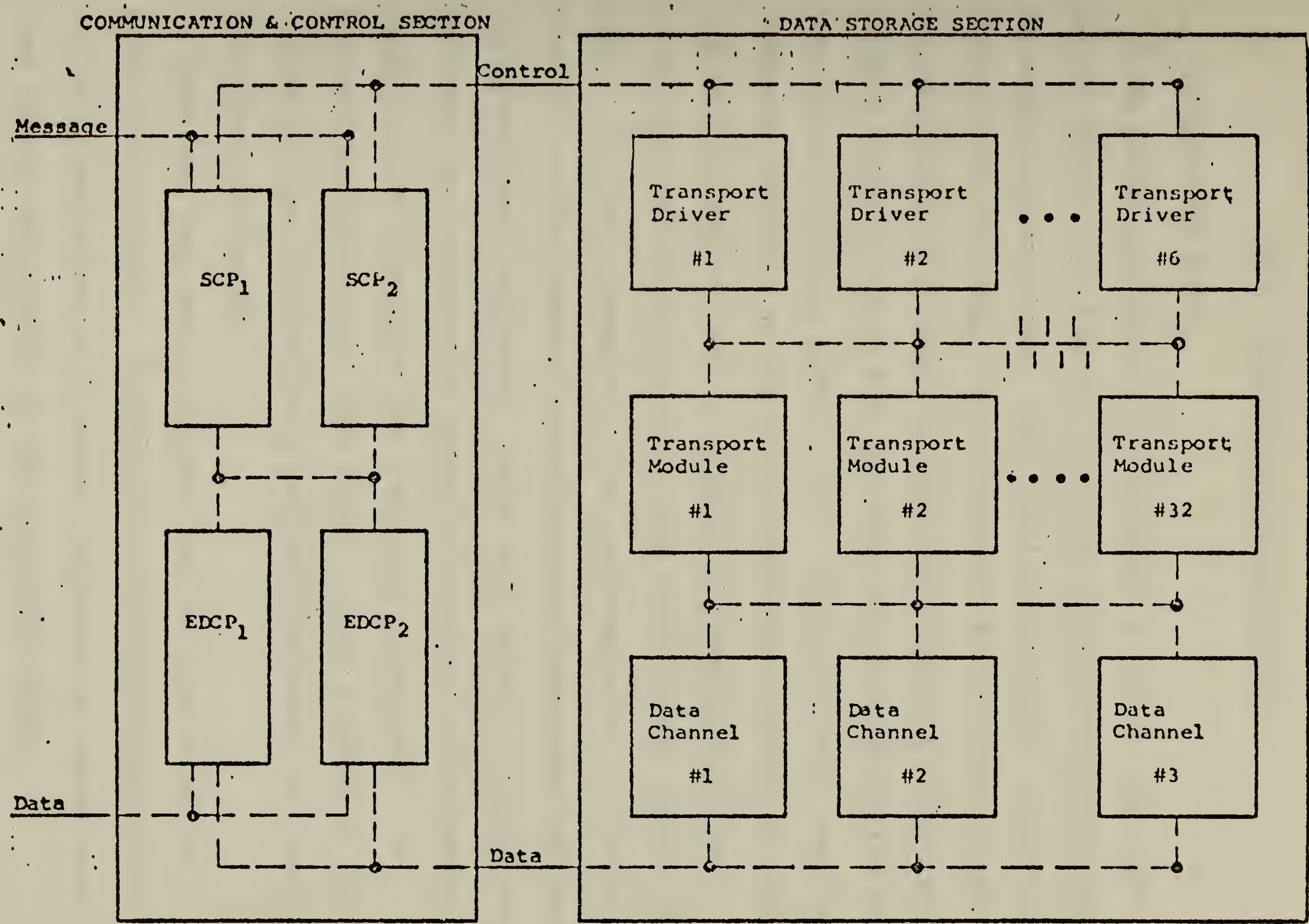


Figure 2 EXPANDED TBM* MEMORY SYSTEM CONFIGURATION

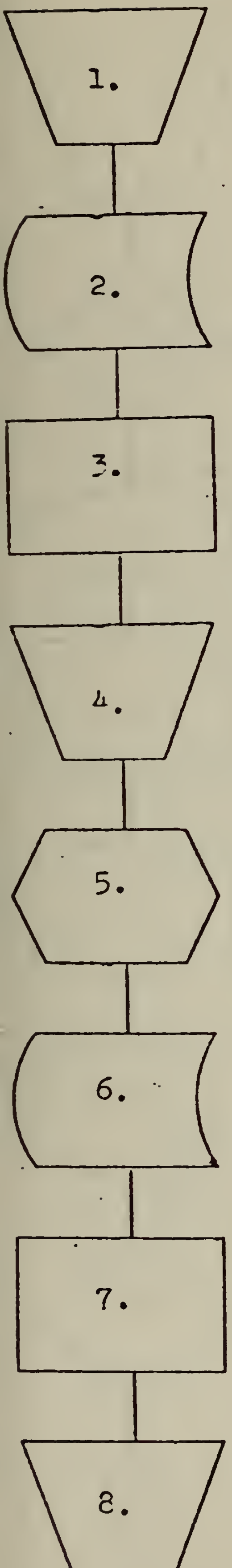
III. TBM* MEMORY SYSTEM MODEL

A. OPERATIONAL MODEL

The Model parallels the management's view of the TBM* Memory System (see Figure 3) and yet incorporates many engineering technicalities. This section describes the development of the TBM model. The System was first examined from the descriptions in Ampex's sales and technical literature. Several trips to Ampex for day-long interviews revealed that the initial proposed system and the adopted system were two different things. By combining technical information from the manuals and the new or updated information from the interviews, an initial logical flow chart of the system was made. This was presented to one representative of the management of the TBM* Memory System research department and one of the development engineers. After many hours of clarifications, the model was approved by Ampex as accurate to the degree necessary for simulation and sensitivity analysis. The approved detailed model (see Figure 4) was similar to and therefore used in place of a program flow chart for developing the simulation model.

The next step was to program the detailed model in a simulation language. In addition to the flow chart, the following variables, with values suggested by Ampex, were used in the development of the simulation model.

Figure 3
OVERVIEW OF THE OPERATION OF THE TBM* MEMORY SYSTEM



1. Host sends data set requests to TBM queue by job priority.

2. SCP looks up address, builds each topplate's search queue by data set location on tape; host puts jobs on wait queue.

3. TBM searches tape and prepares for read.

4. Read from tape through data channel and EDCP to disc.

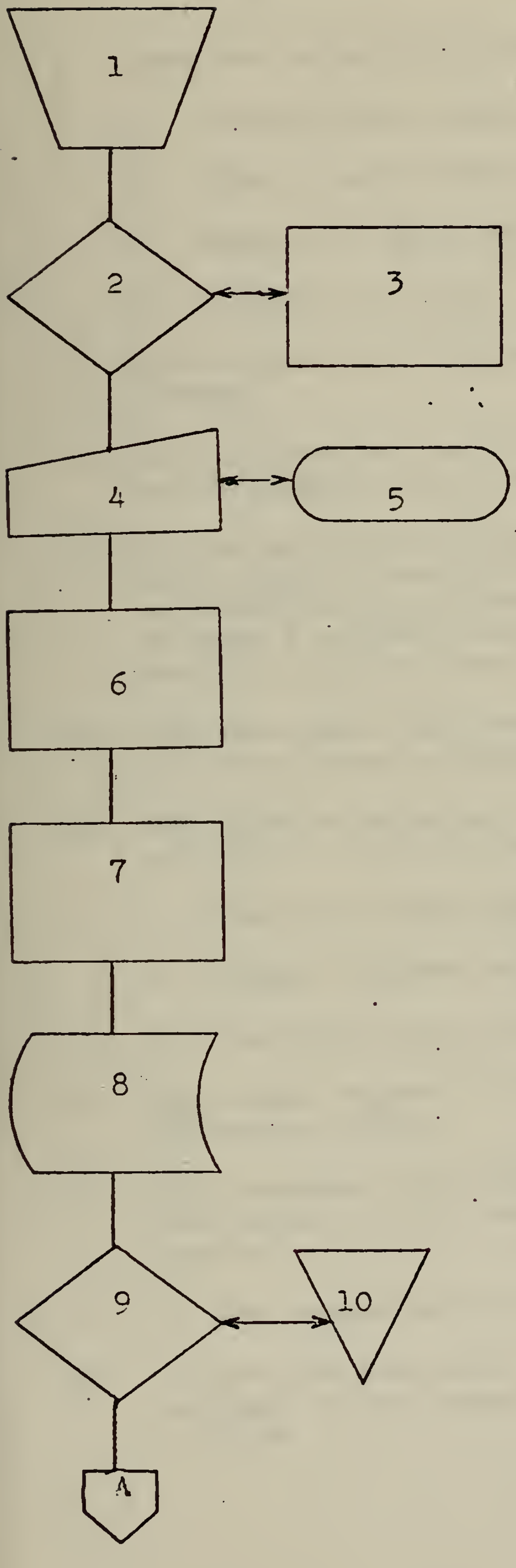
5. Host processes jobs.

6. Host outputs write jobs to queue and signals SCP.

7. SCP searches and prepares for write.

8. Write from disc through EDCP through data channel to tape. Verify and tally.

Figure 4
LOGICAL FLOW CHART OF TBM* MEMORY SYSTEM



1. Input to host

2. Is requested data set located on a device within host system.

3. Assign device type.

4. Send data set requests to SCP.

5. User program goes on inactive status on host queue.

6. Build job input queue. Arrange by priority.

7. Send requests, highest priority first.

8. Search master file for data set.

9. Is data set on-line?

10. Send a message to mount correct tape.

Figure 4 (Continued)

EXPLANATION OF STEPS THROUGH THE TBM* MEMORY SYSTEM

1. Jobs are received by the host with the following parameters:

   A. Interarrival mean and distribution.

   B. Size of each data set request.

   C. Number of data sets per job.

   D. Priority of job.

2. Is the data set located on any device within the host's system?

3. If the data set is resident in the host system, then the host accesses the data set.

4. If the data set is not in the host's system, then the host signals the TBM SCP over a control link (similar to having a write-to-operator request on the console to have a certain tape mounted). The data set requests and job priority are sent to the SCP.

5. The host puts the user program on an inactive status and does other processing.

6. SCP accepts data set requests and queues them by priority. The SCP builds the queue of requests until:

   A. Host finishes read/initiate.

   B. Queue reaches maximum number.

   C. Maximum time limit exceeded by a request.

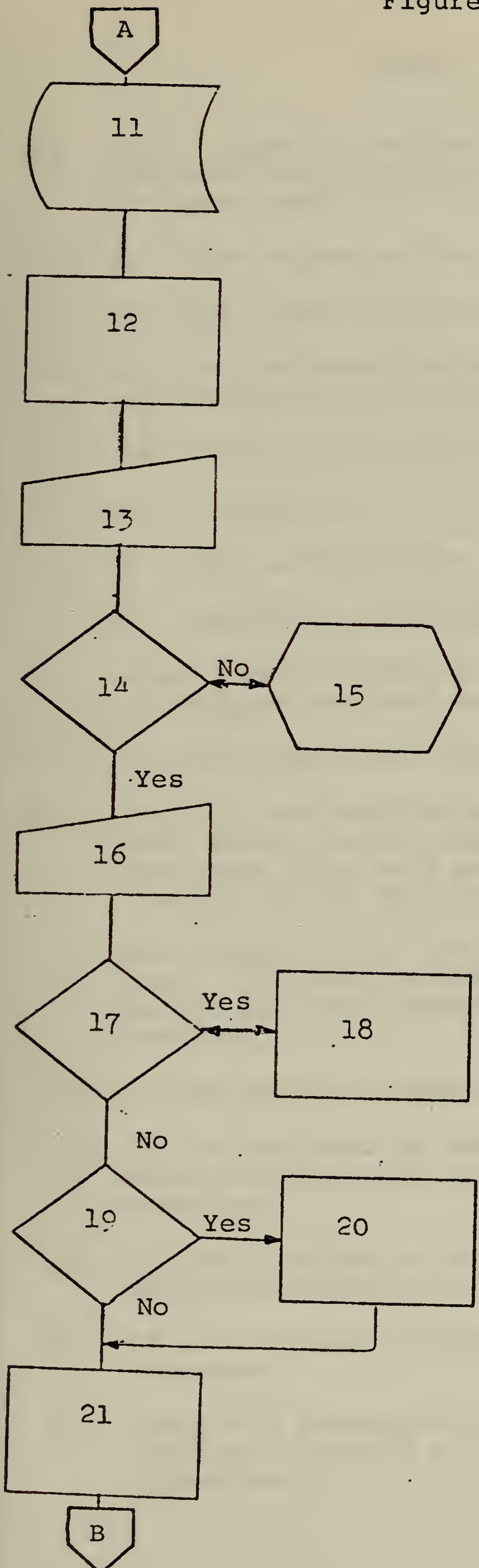
7. SCP sends requests with highest priority to the file directory first.

8. SCP searches the master file directory which contains the status of all data set recorder in the TBM* memory system.

9. Is the requested data set on line?

10. If the TBM tape has been taken off line, then send a message to the operator to retrieve the tape and put it on line.

Figure 4 (Continued)



11. Search on-line file directory. Retrieve the data set description block.

12. Build search queues, one for each topplate, by ascending order of addresses. Wait until N requests arrive.

13. SCP selects queue with N requests and sends to host disc space requirement.

14. Is space available on disc?

15. TBM initiates write or waits for host to release space.

16. Host sends to SCP where to store data on disc. SCP initiates search.

17. Is requested topplate busy?

18. Process other requests.

19. Are all transport drives busy?

20. Wait until a transport drive is free.

21. Connect a transport drive and the correct topplate.

Figure 4 (Continued)

11. If the tape is on line, then the search continues until the data set discrimination block is retrieved. It includes the:

    A. Tape volume serial number.

    B. Disc space required.

12. All the requests for each topplate are queued by ascending order of addresses.

13. The SCP selects a topplate with a queue:

    A. Of length N.

    B. Over maximum time limit.

    C. Containing the highest priority requests.

    Also, the SCP sends to the host the disc space required to contain the data set (multiples of TBM blocks).

14. Is sufficient disc space available?

15. If it is not available, then TBM cannot initiate a read. Disc space will be released by the host in the case of read-only data sets and, if necessary, TBM can initiate a write of the write queue which would free disc space.

16. Host sends to SCP the location where to put the data on disc. SCP must be ready to have the channel simulator modify the home address of the data during the data transfer.

17. Is the topplate busy with previous requests?

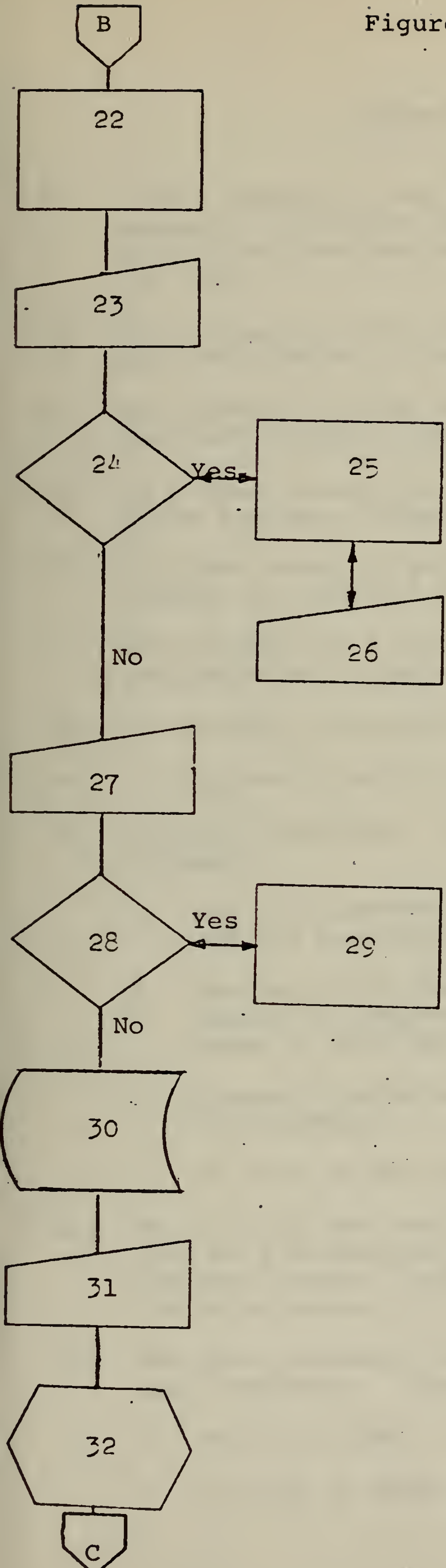
18. If the topplate is busy, the requests for that topplate wait, but parallel processing of other topplates continues.

19. If the topplate is not busy, the SCP assigns the first available transport drive.

20. If all transport drives are busy, then wait until one is released.

21. Connect a transport drive and correct topplate. (This can take from 0.5 to 1.9 seconds depending upon the situation.)

Figure 4 (Continued)



22. Search and preposition tape for read.

23. Transport drive signals SCP ready to read.

24. Are all data channels busy?

25. Transport drive and topplate stay connected and wait for data channel.

26. SCP does other processing.

27. SCP connects transport drive and topplate to data channel.

28. Is disc controller busy?

29. Wait until it is available.

30. Read from tape through data channel through EDCP to disc.

31. SCP to host data available SCP disconnects TD, EDCP, DC and topplate.

32. Host processes data.

Figure 4 (Continued)

22. Search begins at the lowest address of the queued requests. Once the address is found, then the transport drive pre-positions the tape in preparation for the read.

23. The transport drive sends a signal to the SCP when the pre-positioning is complete.

24. The transport drive and topplate logically become one unit and together they request a data channel.

25. If a data channel is not available, then the transport drive and data channel wait.

26. SCP does parallel processing while waiting for a data channel to signal its availability.

27. SCP connects the transport drive and topplate to an available data channel.

28. Is the disc controller busy?

29. If not, wait until it is available.

30. Initiate the read. The data travels from the topplate through:

    A. The data channel for error detection and correction and bit conversion,

    B. External data channel processor which acts as a buffer to compensate for the different transfer rates of disc and TBM* memory system,

    C. Channel simulator to change the disc home address, if necessary,

    D. To disc in the space designated by the host.

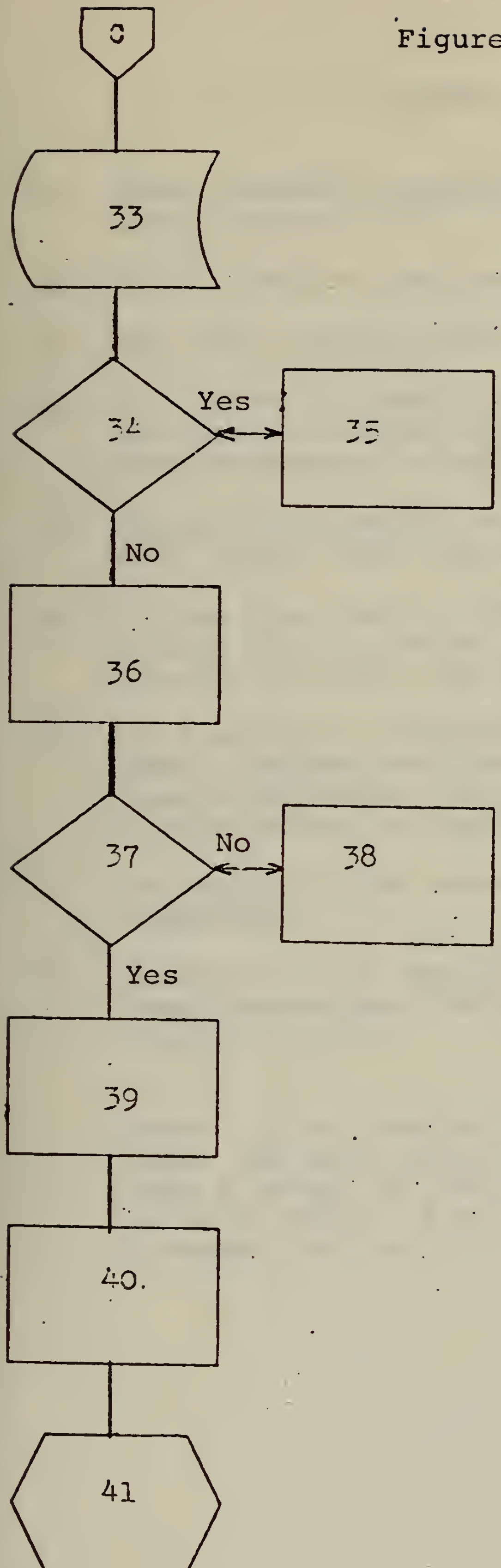
31. SCP signals the host that the data is now available on disc at a designated area. The SCP disconnects the transport drive, the external data channel processor, the data channel, and the topplate.

32. The host processes the data. The host processing time is user dependent. This is not of concern to TBM except:

    A. Write to TBM.

    B. Filling of disc space.

Figure 4 (Continued)



33. Host finished processing outputs data to disc.

34. Was request for read only?

35. Release disc space.

36. Queue writes on disc by tape location.

37. Is more disc space needed?

38. Hold write queue until inactive time, then write.

39. Force a write of queue onto a write tape. Verify and update tally.

40. Release disc space.

41. Hold write tape until inactive time, then write to correct location from write tape. Note: If data set is needed before update from write tape, a search of write tape at 83 IPS is possible.

Figure 4 (Continued)

33. Host finishes processing jobs and writes to disc, if appropriate.

34. Is the request for read only?

35. The host releases the disc space.

36. If writing to TBM*, the queue arranges all write requests on disc and either writes them now or later depending upon management's policy.

37. If more disc space is needed even after releasing read only space, then initiate a write.

38. If the disc space is not needed just now and the processing load is large, then wait for an inactive time of the day to write the data from disc to TBM*.

39. If a write is necessary to free disc space, then a faster method for clearing the disc is to write to a dump tape rather than updating the original files during a busy period of the day.

40. Release the disc space after the write and verify is completed.

41. Hold the write tape until an inactive period of the day, then update the TBM* master tapes without tying up host, discs, or EDCP.

Note: If an updated data set is needed from the write tape, the host should keep a write tape file directory. Then a search in the usual manner could be made. This is also useful if the write dump tape is to be used as a backup for security.

1. Host Controlled Variables

		Batch
1)	Job Arrival Rate	Ave 80/hr, range 60-120 hr; CP (time sharing)-ave. 20/hr; Initial sign on - 20/hr; Range 0 - 25/hr
2)	Job Arrival Distribution	Poisson - with above average; constant - (i.e., 1 job/min); Immediate - 20 jobs arrive together
3)	Job Priority	The priority of a job only influences its relative position when the job is released into the TBM* System; after being released into the system the job's priority has no effect.
4)	Number of Data Sets/Job	CP - 2. Batch - 2, 5, 8. If 5 or 8 data sets are requested then usually 2 of them are very large.
5)	Size of Each Data Set Request	CP - Average 512 K bytes. Batch - Average 2.5 megabytes; Range 128 K bytes - 30 megabytes; See Figure 5.
6)	Number of Disc Controllers	Single Access - 1 controller Dual Access - 2 controllers
7)	Number of Disc Drives	Host dependent

2. TBM* Controlled Variables

1)	Number of Transport Drives Available	Minimum System of 1; Maximum System of 6
2)	Initial Tape head location	Mid tape; Beginning of tape; Random location. The initial bias was eliminated before gathering statistics on each run.

3) Search speed is dependent upon search distance. See following section on search speed and times.

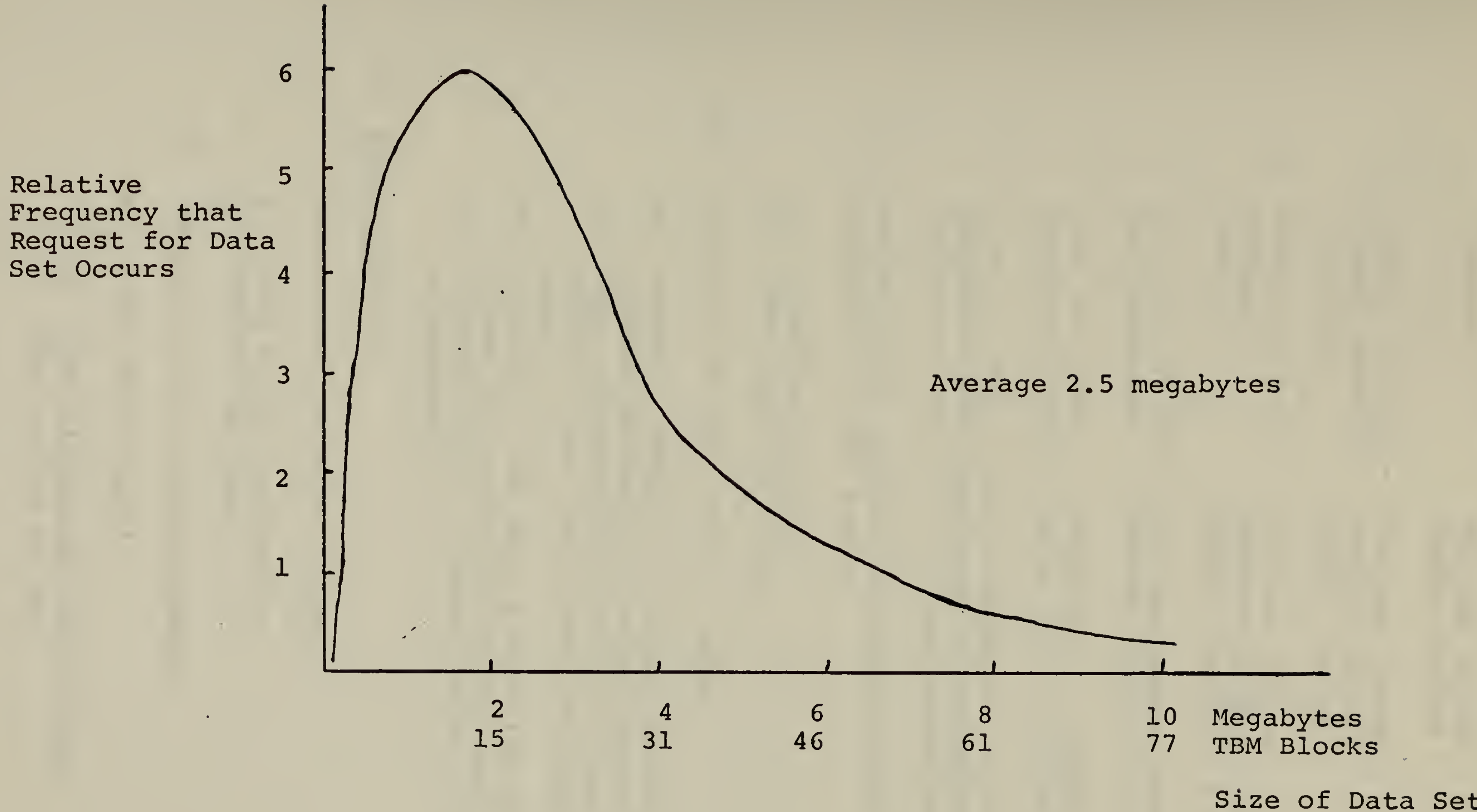


Figure 5 DATA SET SIZE FOR BATCH MODE

4)	Number of Data Channels	Minimum System - 1 Maximum System - 3 Number tested - 1, 2
5)	Number of Supervisory Control Processors	Minimum System - 1 Maximum System - 2 Number tested - 1
6)	Number of Dual Transport Modules	Minimum System - 1 Maximum System - 36 Number tested - 1 - 36
7)	Per cent of Read-only Requests	Any percentage possible. Tested 75%
8)	Time for File Directory Search	Average 0.1 seconds. Range 8.5 - 1000 milliseconds.

9) Time for search and preposition of the tape is dependent upon distances to be searched. See following section on search speeds and times.

10) Job priority queuing techniques before releasing to TBM System.

   a) Length of queue.
   b) Time spend in queue.

11) Location of Data Sets on Tape:

   a) Random - Assume all types of data sets are located uniformly over the whole tape.
   b) Size - Arrange data sets by size. The larger the data set, the closer to either end of the tape.
   c) Activity - the more active the data set, the closer to the middle of the tape.
   d) Size/Activity - A combination of the above arrangement.

3. <u>Write Options</u>

   1) Replacement of Write Updates

      a) Into final destination.
      b) Onto Write Tape.

   2) Release time of write updates.

      a) Queue Write until

         1) Disc limitation force write
         2) Slow period of day

b) Send to TBM System as released by host.

B. SIMULATION MODEL

1. Programming Language

The model is written in General Purpose Simulation System language (GPSS). The complete program is in Appendix I. The model is programmed to include the engineering technicalities while following the management's operational flow chart as described previously in Figure 4. GPSS is an event-oriented discrete simulation language. All the model's events are paralleled by GPSS blocks. To make the model more accurate, the time units of the model were tenths of seconds.

2. Search Speeds

The search time for locating a particular TBM block is based on the following assumptions. If the topplate is waiting for the search signal, then the tape is not moving. When the signal to search at 1000 IPS arrives, there is no delay waiting for DTM (Dual Transport Module) to connect to the TD (transport drive) since that operation was done previously. The acceleration to 1000 IPS begins and for purposes of the simulation is assumed to be constant. The time required to reach 1000 IPS is 1.2 seconds. The error introduced into the model by the assumption of constant acceleration is trivial when compared to search times. The statistics for deceleration are the same as those for acceleration.

a. Mathematics of Search Time

A constant acceleration from 0 to 1000 IPS, or deceleration from 1000 IPS to 0, in 1.2 seconds means that 600 inches were covered at an average rate of 500 IPS.

Average Velocity = 1000 IPS ÷ 2 = 500 IPS;

Distance Covered = 1.2 seconds X $\frac{\text{500 inches}}{\text{seconds}}$ = 600 inches.

Since 1 TBM Block = 0.9752 inch or

1 inch = 1.02543 TBM Blocks,

Distance Covered = 600 inches X 1.02543 $\frac{\text{Blocks}}{\text{inch}}$ =

615.26 Blocks

So 615 blocks are travelled in reaching 1000 IPS or decelerating from 1000 IPS to stop. Every search has an acceleration period and a deceleration period. If the search speed reached 1000 IPS, then 1230 blocks were covered at some speed less than 1000 IPS during acceleration and deceleration periods.

Since acceleration is constant, it can be computed:

$$\text{ACC} = \frac{\text{1000 IPS}}{\text{1.2 seconds}} = 833.33 \text{ IPS}^2$$

The maximum velocity attainable is 1000 IPS. If the search distance is less than 1230 blocks, then 1000 IPS will not be reached and the search time is calculated as in the following example.

Example: Search distance = 1000 blocks. Accelerate for 500 blocks and decelerate for 500 blocks.

Maximum velocity will be less than 1000 IPS and is computed as follows:

Velocity Maximum = Constant Acceleration Rate X Time of Acceleration

The tape is accelerated for 1/2 of the total time of travel and decelerated the other half. (There was no period of constant velocity since 1000 IPS was not reached.)

$$\text{Velocity Maximum} = \text{Acc. X } \frac{\text{time}}{2}$$

$$= 833.33 \text{ X } \frac{\text{time}}{2}$$

The distance travelled equals 1000 blocks, according to this example. The distance travelled is also equal to average velocity X time of travel. Average velocity equals velocity maximum ÷ 2: (Note: This is not true if 1000 IPS attained.)

$$V_{average} = V_{max} \div 2.$$

Substituting and solving for time gives:

$$D = \frac{1}{4} \text{ Acc X } T^2$$

$$T = \sqrt{\frac{4D}{ACC}} \quad = \sqrt{\frac{4D}{833.33}} \qquad D < \quad 1230$$

therefore, when search distance is less than 1230 blocks, the time will be computed from the above equation.

When the distance is greater than 1230 blocks, then 1000 IPS will be reached. The acceleration time plus deceleration time equals 2.4 seconds and the distance covered is 1230 blocks. Any search distance greater than

1230 blocks will have an acceleration period of 1.2 seconds, a deceleration period of 1.2 seconds, and a constant velocity period determined by search distance, D. (See Figure 6).

Time to search greater than 1230 blocks is given by:

$$\text{Time} = 2.4 + \frac{D - 1230}{1000}, \quad D \geq 1230 \text{ Blocks}$$

Note: The error induced by the assumption of constant acceleration and deceleration is trivial when compared to search times. Appendix C has an explanation of search speeds.

3. Major Time Periods

The simulation model is built for a typical case borrowed from Ampex* under peak loads. Any special situation can be included with minimal changes but attempting all special situations is infeasible. The major time periods of the model are as follows:

I. Jobs requesting data sets enter TBM* and wait until:

   A. Enough jobs have arrived;

   B. Some job has been in queue a maximum time limit. While waiting, jobs or data set requests are arranged in a queue by priority. Each job has one or more data set requests.

II. Transactions or data set requests enter queue for a particular topplate and wait until:

   A. Enough requests for that topplate exist (i.e., 10);

   B. Some request has been in over maximum time limit.

III. Transactions for a topplate enter a queue waiting for a transport drive. After being allocated a transport drive, there is a short delay for connections.

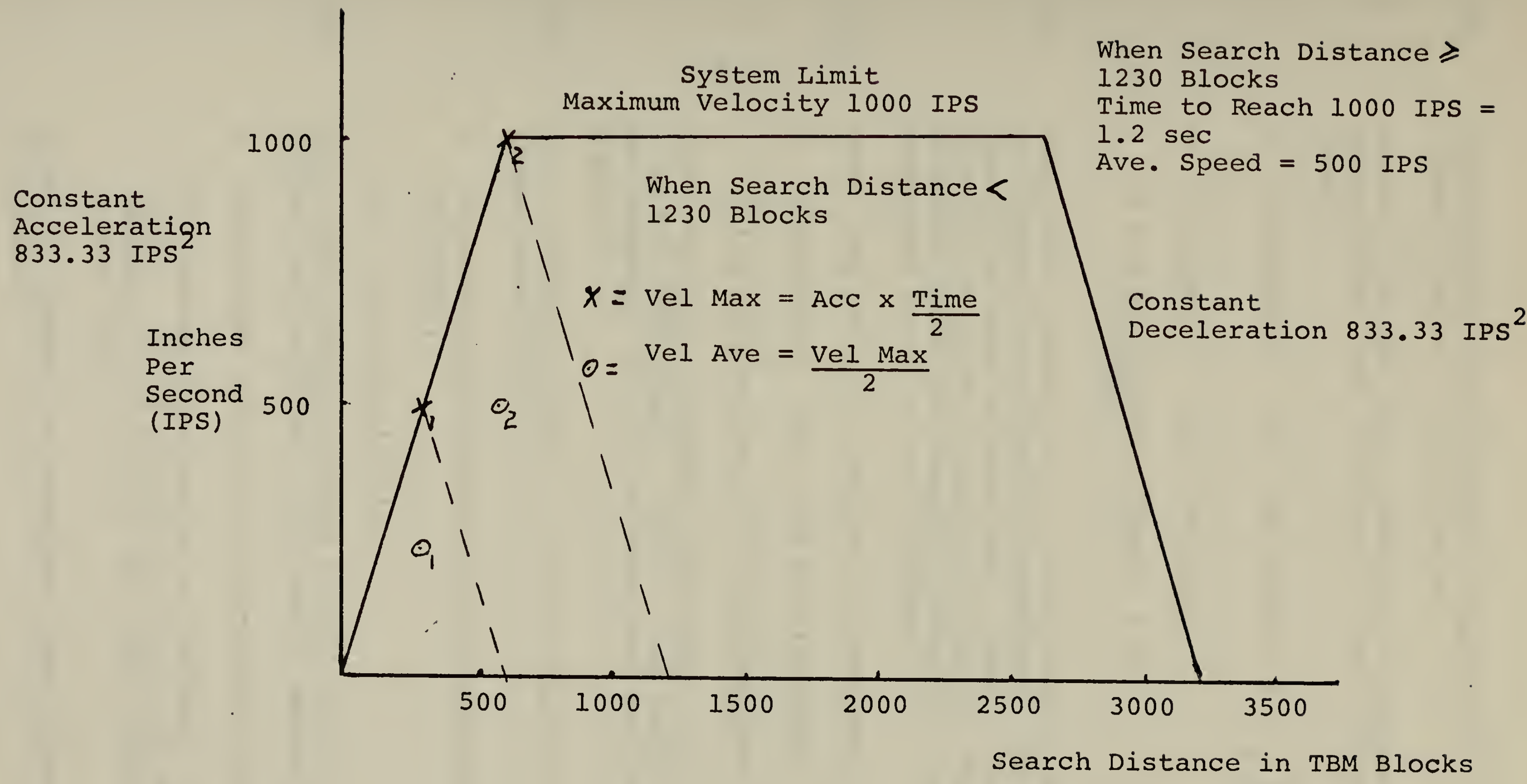


Figure 6 MODEL SEARCH SPEEDS

IV. Transactions for topplate sequentially advance through Steps V to VII below.

V. Search time for each transaction assumes constant acceleration and deceleration (see Chapter III-B-2 above).

A. Distance < 1230 TBM blocks, $T = \sqrt{\frac{4D}{ACC}} = \sqrt{\frac{4D}{833}}$

B. Distance $\geq$ 1230 TBM blocks,

$$T = \frac{\text{\# of blocks} - 1230}{1000} + 2.4$$

VI. When each transaction is found then a Data Channel, an External Data Channel Processor, and a disc controller must be obtained. Note: the disc space should have already been allocated. The devices are released after the reading operation, except the transport drive is not released.

VII. Read time for each request is composed of two constants

A. Preposition time, 250 milliseconds;

B. Tally update time, 200 milliseconds.

and a variable dependent upon data set size, i.e., 0.19 sec./TBM block. (Note: the above three steps are repeated for all transactions waiting for that topplate.)

VIII. Host processing time is not of concern to TBM* Memory System except for:

A. Write entries into TBM* (for rewriting updated records);

B. Disc space available.

IX. Write time affects system only if writes are necessary during busy periods. This would happen only if:

A. Management wanted files updated immediately;

B. Not enough disc space is available to hold write queue. It is practical to put all writes on a temporary write tape for later updating of the master file.

X. Updating speeds are dependent upon number of blocks to be updated:

A. Erase: 5 or 83 IPS;

B. Tally: 83 IPS;

C. Verify 5 IPS.

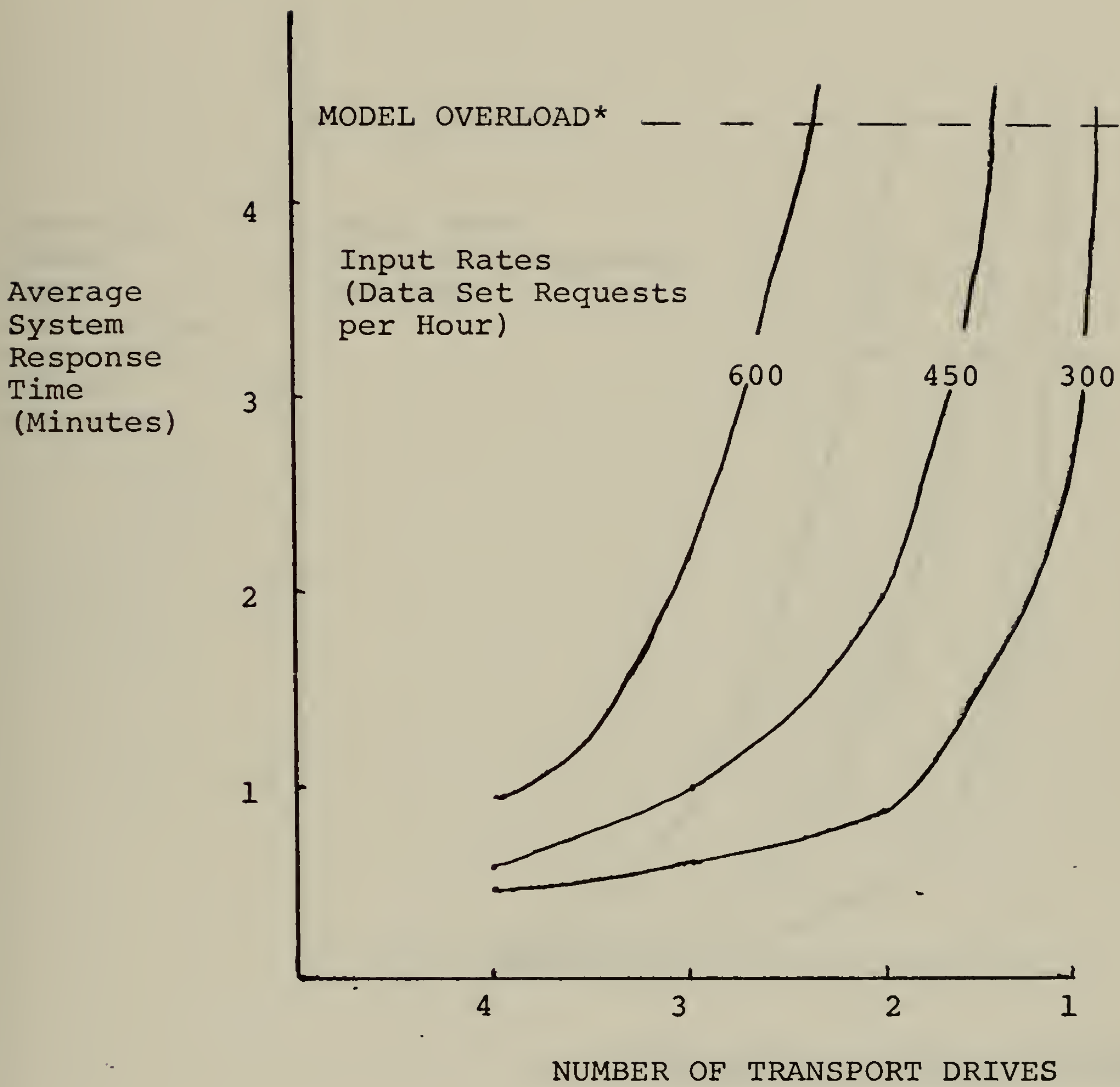
Note: Appendix C has an in depth explanation of hardware time periods.

IV. ANALYSIS

A. SYSTEM RESPONSE TIME VERSUS TRANSPORT DRIVES

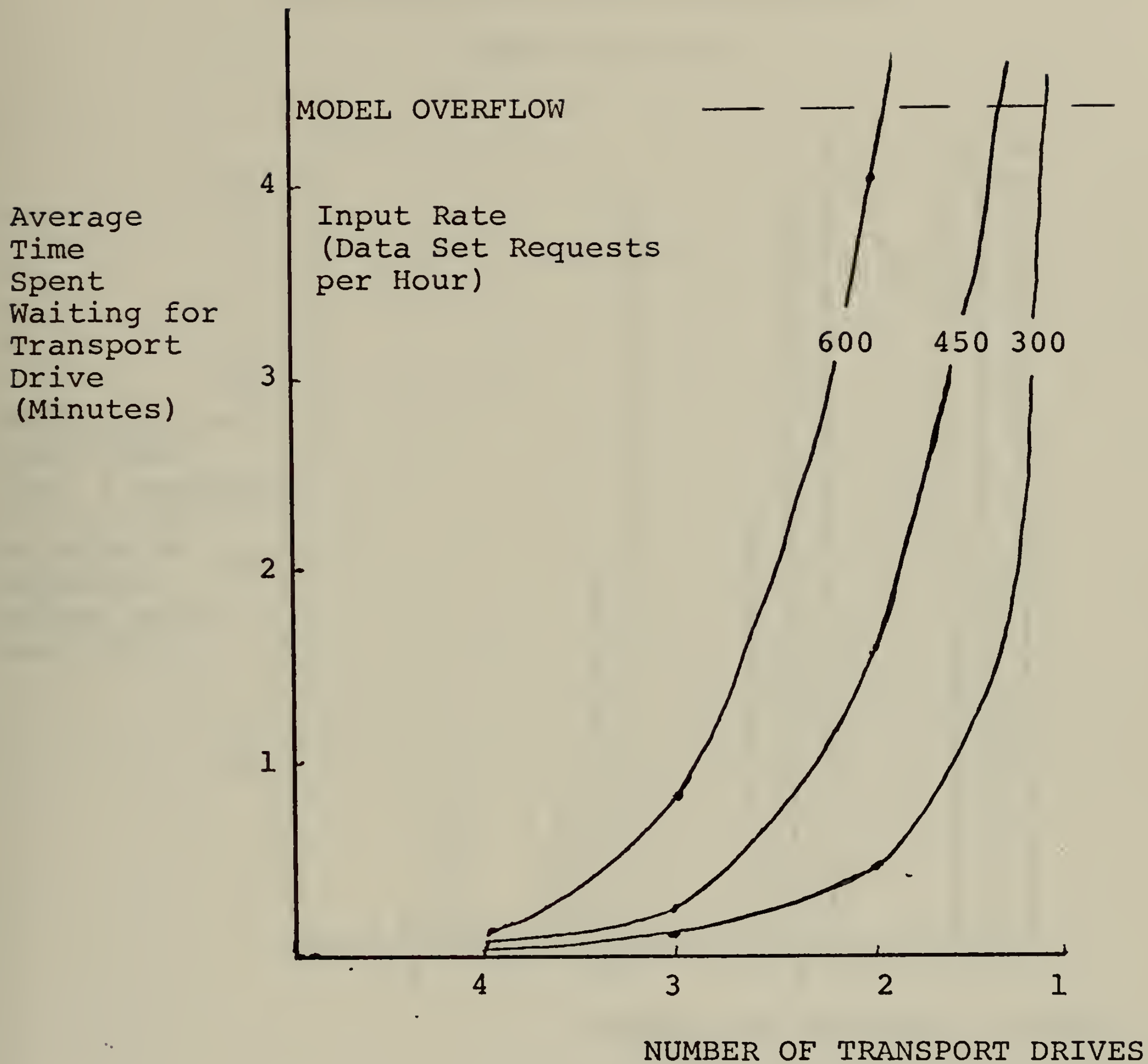
This section describes the average system response time and how it varies with number of transport drives. First, a perceptive conclusion is derived from Figure 7. The average system response time varies inversely with the number of transport drives available, and directly with the input rate. The next step is to find out where and why serious delays are encountered in the system. In examining the delays, the following variables are held constant. 1) Data sets were uniformly arranged over the tape with no bias as to size or activity. 2) The queue minimum level before a search was initiated was one. This tested the demand on the transport drives as the requests arrived. If more requests arrived after the first one and before a transport drive could be secured, then the requests were arranged in sequential order by physical location. The sequential ordering reduces the average tape search time and therefore the average system response time.

The first section will find the delay caused by the non-availability of transport drives. Figure 8 illustrates the average time spent waiting for a transport drive. The similarity of the average system response time in Figures 7 and 8 shows that the system delay is largely due to lack of transport drives. Figure 9 illustrates that as number of transport drives decreases, the percentage of total time



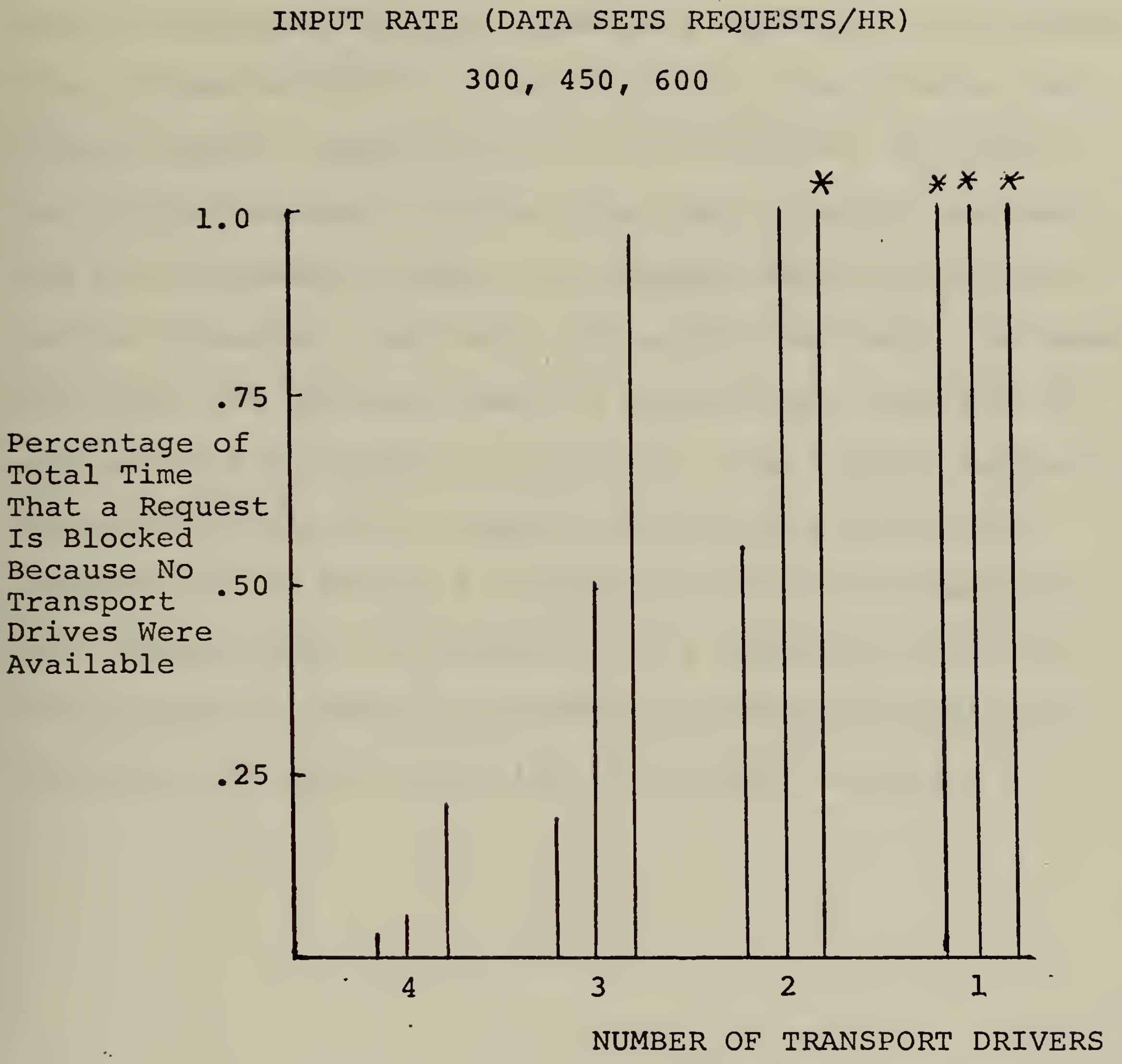
*Model Overflow means requests were arriving faster than they could be processed - Result: Large backlogs of jobs that could not be handled by simulation program.

Figure 7 AVERAGE SYSTEM RESPONSE TIME VS. NUMBER OF TRANSPORT DRIVES



The Average Amount spent waiting for a transport drive varies inversely with number of TD available and directly with input rate.

Figure 8 AVERAGE TIME SPEND WAITING FOR A TRANSPORT DRIVE

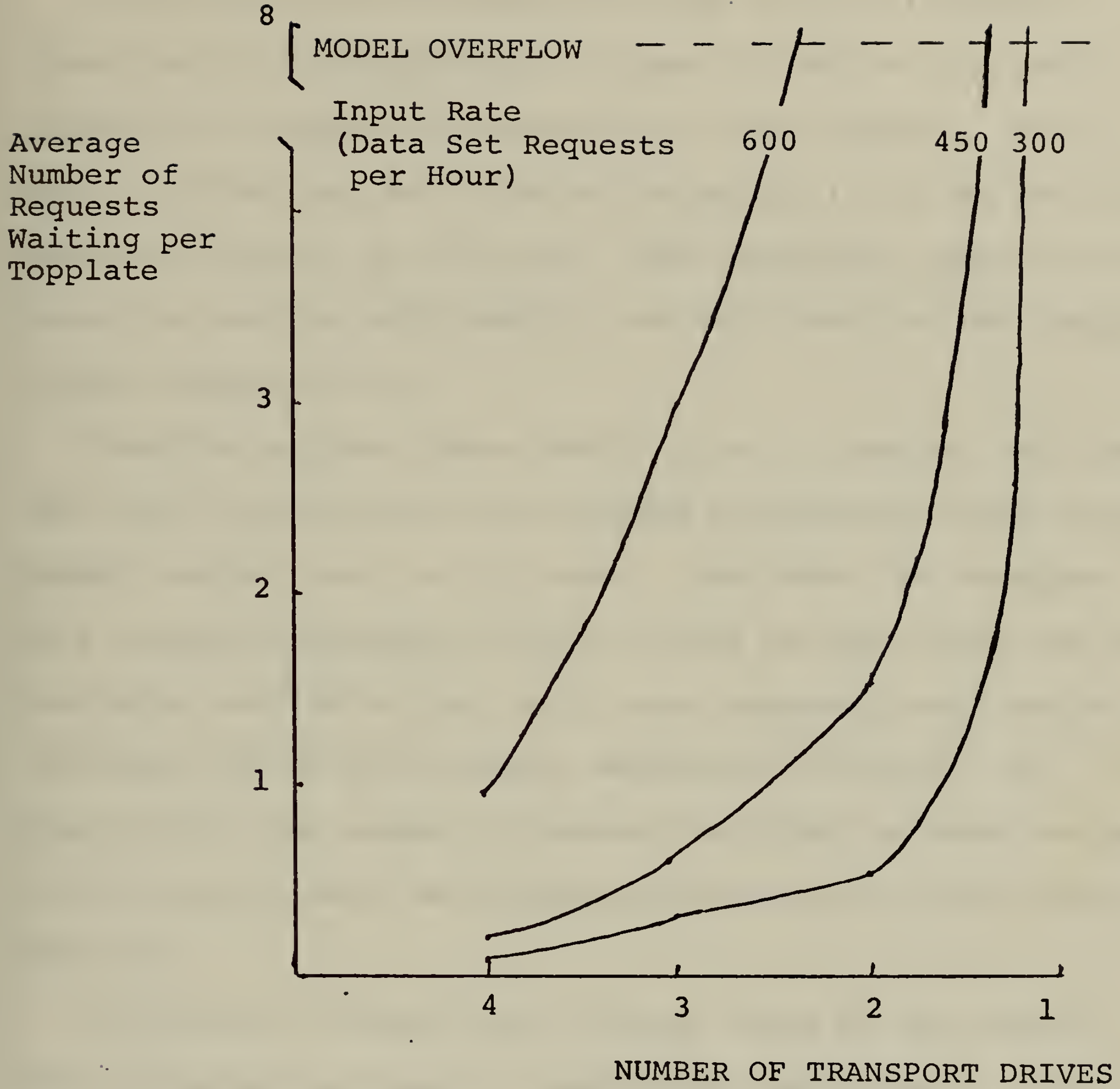


*Model Overflow

This is the percentage of total time that any request was blocked because of non-availability of a transport drive and not the utilization of each transport drive.

Figure 9 PERCENTAGE OF TOTAL TIME THAT A DELAY WAS DUE TO NON-AVAILABILITY OF A TRANSPORT DRIVE

that at least one request is waiting for a transport drive increases. This naturally pushes the average time delay due to non-availability of transport drives up. The higher the utilization of all transport drives, the greater the average system response time. The statistics in Figure 9 are the percentages of total time that a request was waiting for a transport drive. As average delay for transport drives increases, then more jobs arrive and enter the queue. Naturally the average number of requests per topplate increases as illustrated in Figure 10. The average number of requests waiting for a topplate increases because more requests arrive before a transport drive can be secured. The average number of requests for a topplate varies inversely with number of transport drivers available and directly with input rate also illustrated in Figure 10.



The minimum topplate queue length before a search was initiated is 1.

Figure 10 AVERAGE NUMBER OF REQUESTS PER TOPPLATE

B. SYSTEM RESPONSE VERSUS QUEUE LENGTH

This section will attempt to find a logical minimum queue length for topplates. A queue is defined for each topplate to accept all requests for that topplate. As requests arrive they are arranged sequentially by the data set's physical location on the tape. The sequential ordering reduces the average tape search time and therefore the average system response time.

When the minimum queue level is set to greater than one, then that topplate does not request a transport drive until enough requests are in the queue. Even when the topplate does request a transport drive, it has to wait until one is available and during that wait, more requests could enter the queue and be put in their sequential location. In other words, the number of requests batched for each search may be greater than the minimum queue level but never less than it.

The optimal minimum queue length would be the length that minimized the average search time per request. Although the logical minimum queue length as presented later in this thesis may be optimal for some users, in general the optimal will have to be determined by the user for his particular installation. This analysis provides a first approximation for determining the optimal minimum queue length. Referring to Figure 11 will be of assistance in the following analysis. The goal is to try to minimize

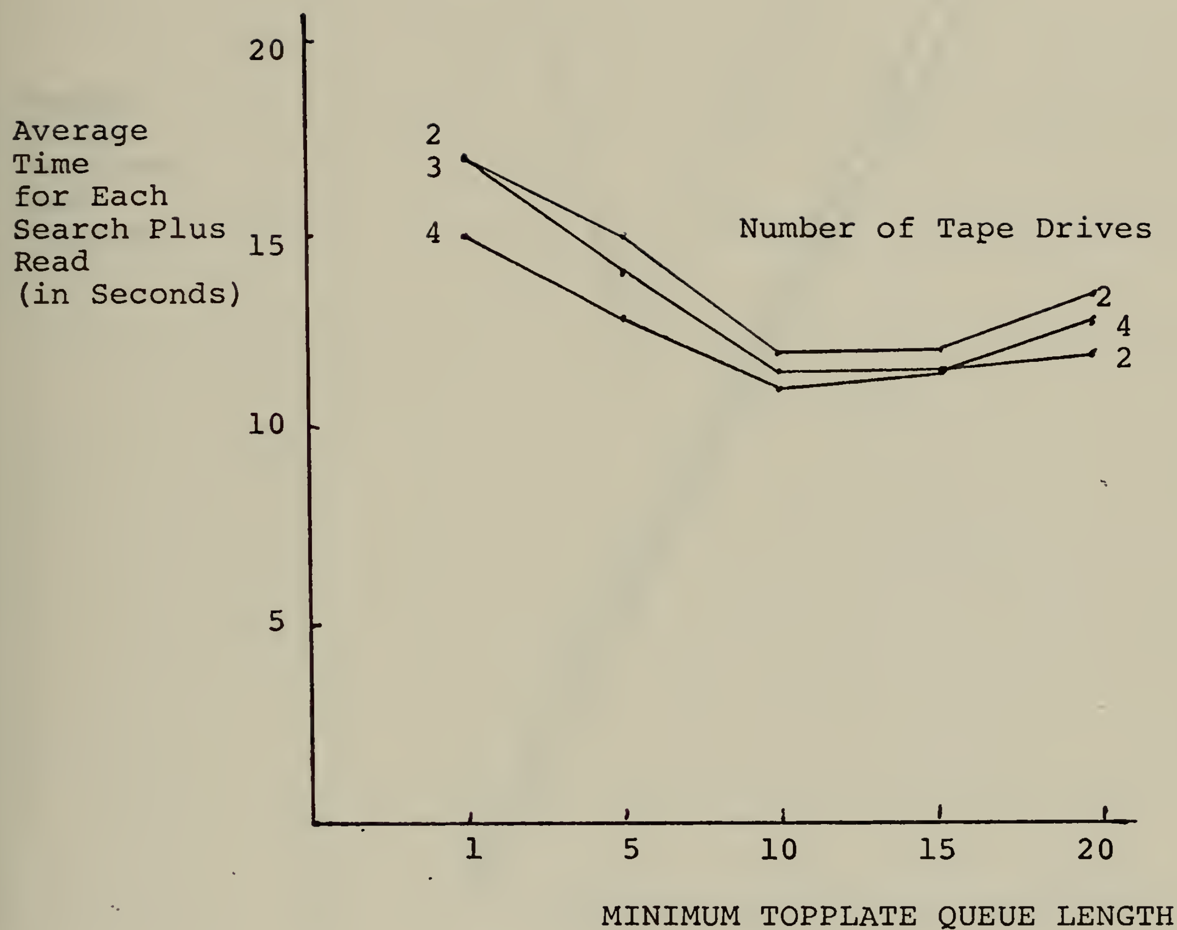
average search time per request after the batch of requests has been released and the searching has been initiated. In order for the batch to be released for searching, a transport drive must be secured. Consequently, the number of transport drives has no effect on the average search time per request. The minimum search time is achieved when the queue is of a length between 10 and 15 requests. The data set size is held constant. Consequently the read time is constant for all queue lengths and only the search time varies with queue length. (See Figure 11). The average system response time is also dependent upon three other factors:

1) Number of transport drives
2) Topplate queue length
3) Input rate.

By holding the input rate constant and varying the other two main factors, the average system response time is sensitive to both the number of transport drives and the topplate queue length, as shown in Figure 12. The linear increase in average response time is due to waiting for the arrival of enough requests to fill the queue to the minimum queue level for the topplate. This conclusion is supported by Figure 13 which shows the average time that a data set request is held in the queue before being released for searching and processing.

This conclusion is valid except when the number of transport drives becomes a critical factor in system response time. In the case of one or two transport drives,

the average time in the system increases drastically. This is due to the average time spent waiting for enough requests to fill the queue for the topplate, as illustrated by Figure 14.

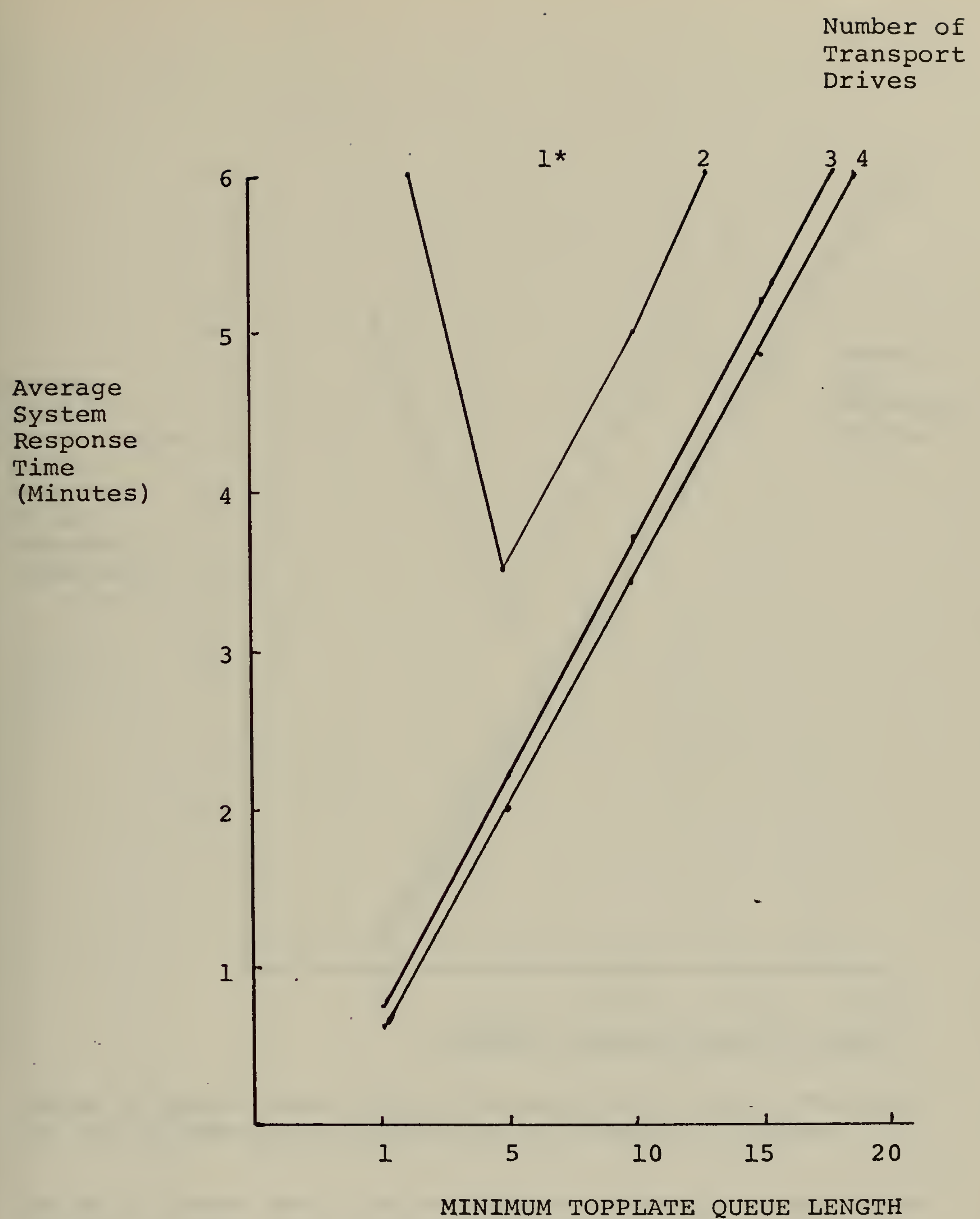


Constant variables input 450 data set requests/per hour

Uniform distribution of data sets on tape

The read time is constant for all queue lengths so only the search time varies.

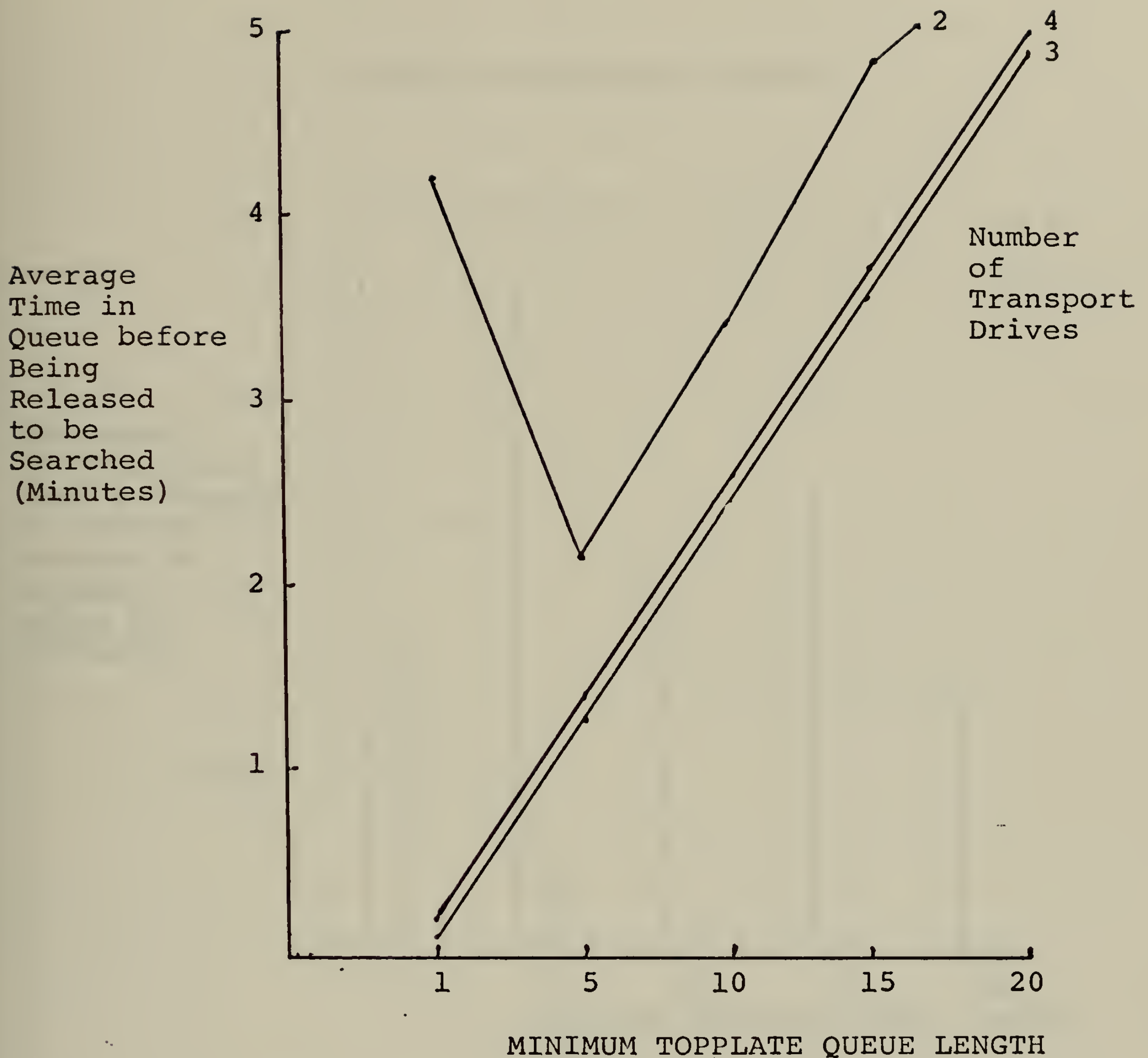
Figure 11 AVERAGE TIME TO FIND AND READ EACH DATA SET REQUEST



Most of time is spend waiting for enough requests to arrive.

*With 1 transport drive the model overflowed.

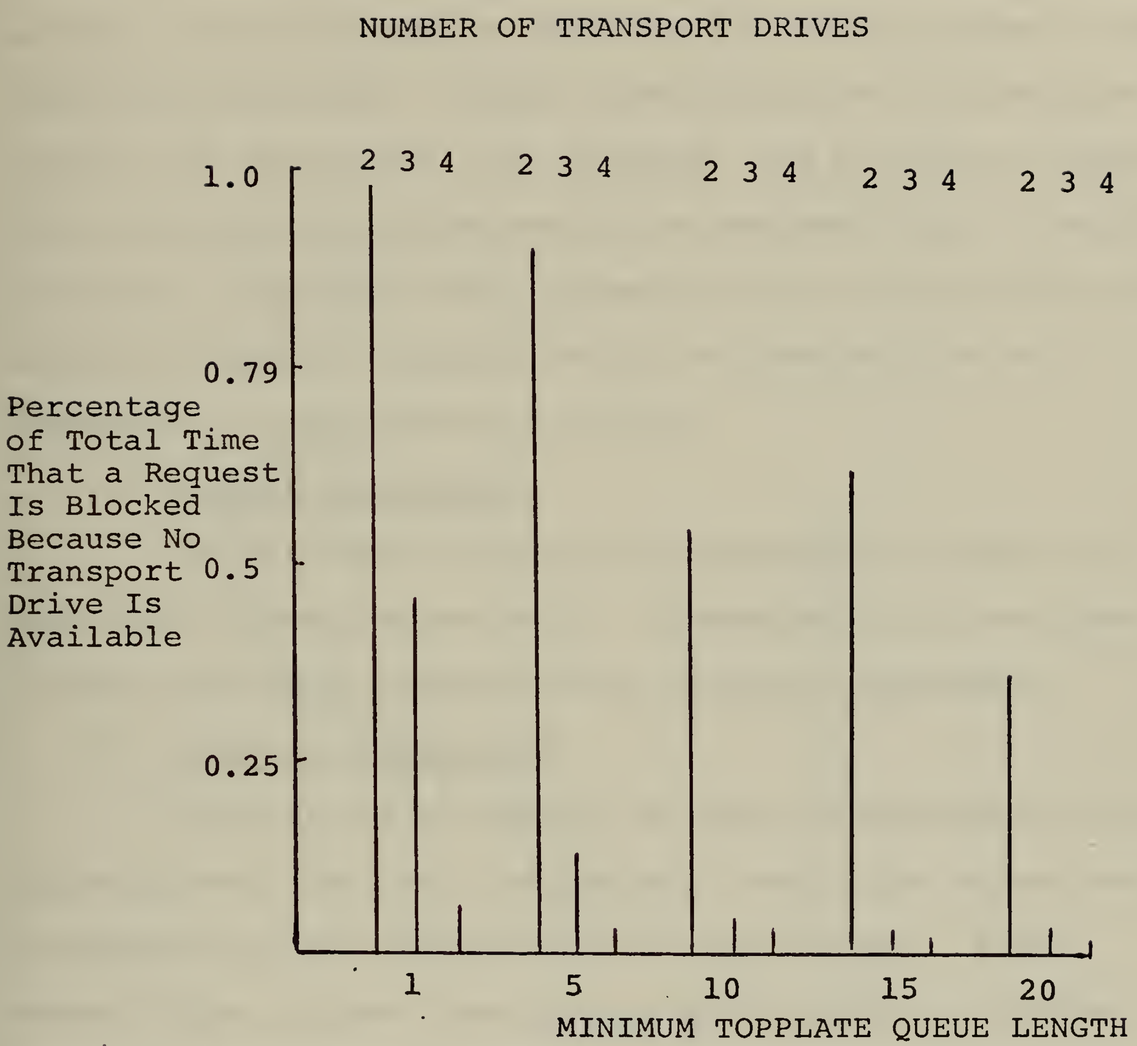
Figure 12 AVERAGE SYSTEM RESPONSE TIME VS. MINIMUM TOPPLATE QUEUE LENGTH



Delay in queue when only 2 transport drives are available is due to the non-availability of transport drives.

Delay in queue when 3 or 4 transport drives are avaialble is due to wait for enough jobs to arrive.

Figure 13 AVERAGE TIME IN QUEUE BEFORE BEING RELEASED



Note: This is percentage of total time that any request was blocked due to the non-availability of a transport drive and not the % of time a TD is busy.

Figure 14 PERCENTAGE OF TIME A REQUEST IS BLOCKED DUE TO NON-AVAILABILITY OF A TRANSPORT DRIVE

C. SYSTEM RESPONSE TIME VERSUS DATA SET ARRANGEMENT CRITERIA

The file management policies can affect the system response time. A careful analysis of three different methods or data set arrangement on TBM Tape will be described in the following paragraphs. These three arrangement methods are uniform (or sequential), by activity, and by data set size. Since the data set arrangements are mathematically formulated, no claim is made that they represent any one particular installation and the analysis should be used only as an indication of many possible results.

1. Uniform Arrangement

The data sets are placed in sequential order on a first-come first-located basis. Consequently all sections of tape have equal probabilities of being referenced.

2. Activity Arrangement

The data set's location on tape is determined by its activity over the past time period. Every time a data set is referenced, the tally track is also updated. After several time periods, the data sets on the tape would be rearranged according to activity during those time periods. The more active the data sets are, the closer they are to the middle of the tape, (See Figure 15). The idea is to put the most frequently accessed data in a location where they can be accessed most quickly. This idea is very appealing to installations that follow the adage - "90% of the accesses are to 15% of the data base."

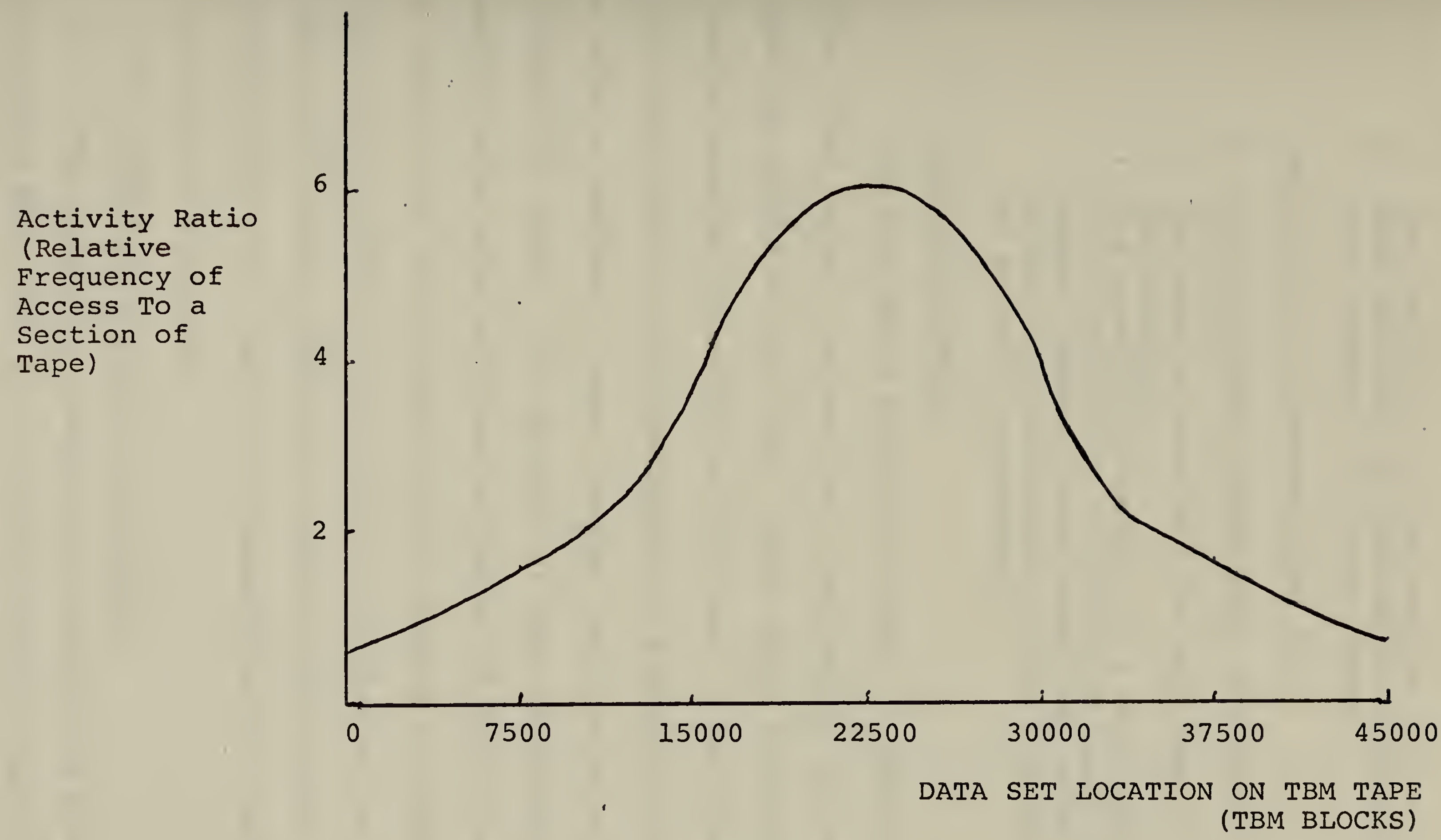


Figure 15 ACTIVITY ARRANGEMENT OF DATA SETS ON A TAPE

The principle advantages of the arrangement by activity is:

1) The reduction in the average search time;
2) Data sets may be rearranged independent of the host computer, in slack times.

The principal disadvantages are:

1) The necessary bookkeeping in order to arrange by activity.
2) The need to re-order the data sets on a tape periodically.
3) The possible placement of a very large data set in the middle of a tape which means greater search times to many other data sets. This result may cancel and reverse any savings gained by activity arrangement.

3. <u>Size Arrangements</u>

The data set's location on tape is determined by its size. The larger the data set, the farther it is located from the middle of the tape. Thus, small data sets are located towards the middle of the tape -- See Figure 16. The logic behind this arrangement is based on activity over a certain section of tape. Assume the data sets have a uniform hit ratio. The section of tape containing the smaller data sets will have a higher hit ratio per length of tape. The idea of putting the highest hit section of tape in the middle of the tape is to minimize the search times and thereby reduce the average access time. The major problem with this idea is the assumption of a uniform hit ratio on all data sets.

4. <u>Size and Activity Arrangement</u>

The purpose of combining activity and size criteria to arrange data sets is to avoid putting a large data set in

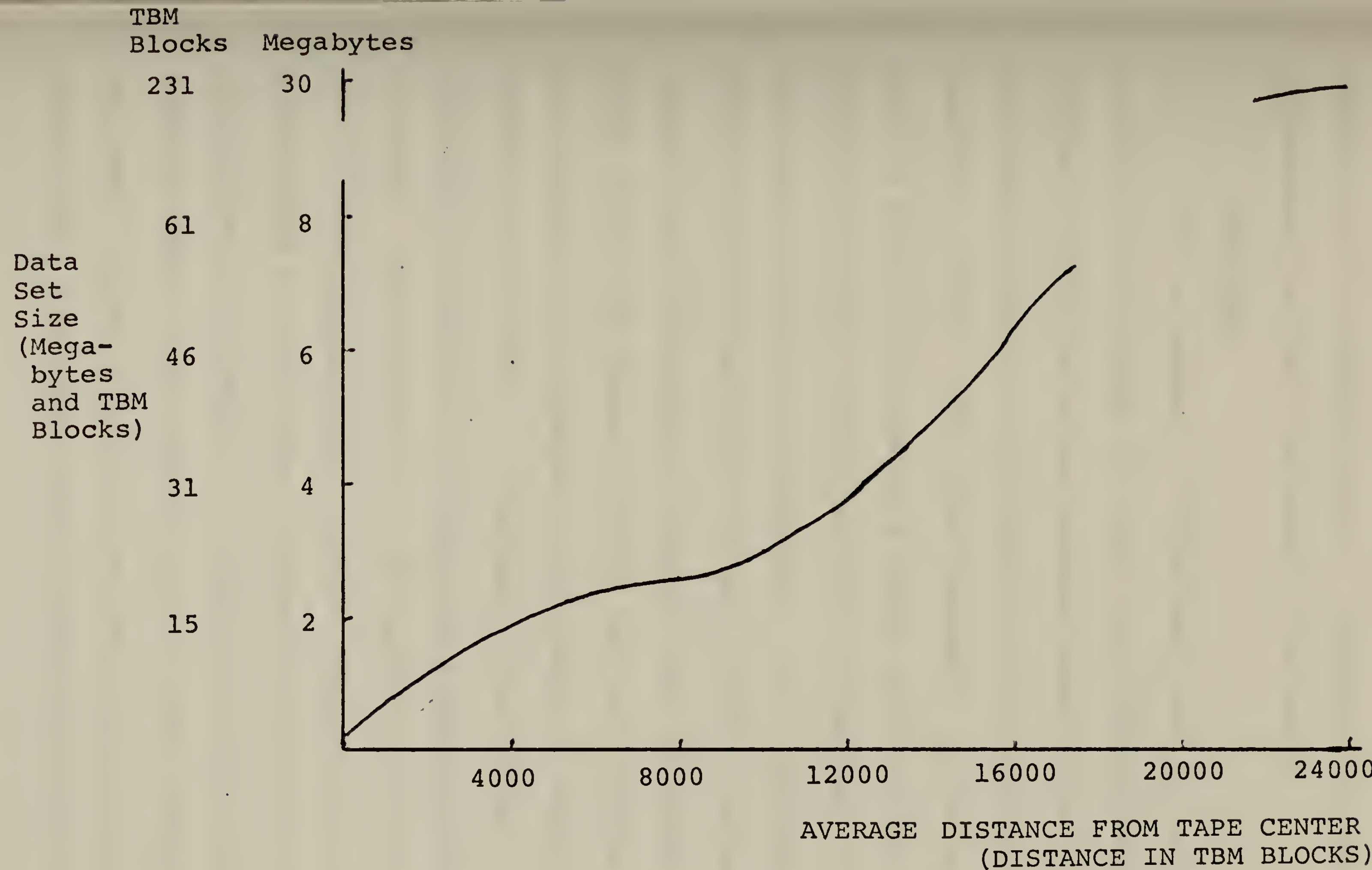
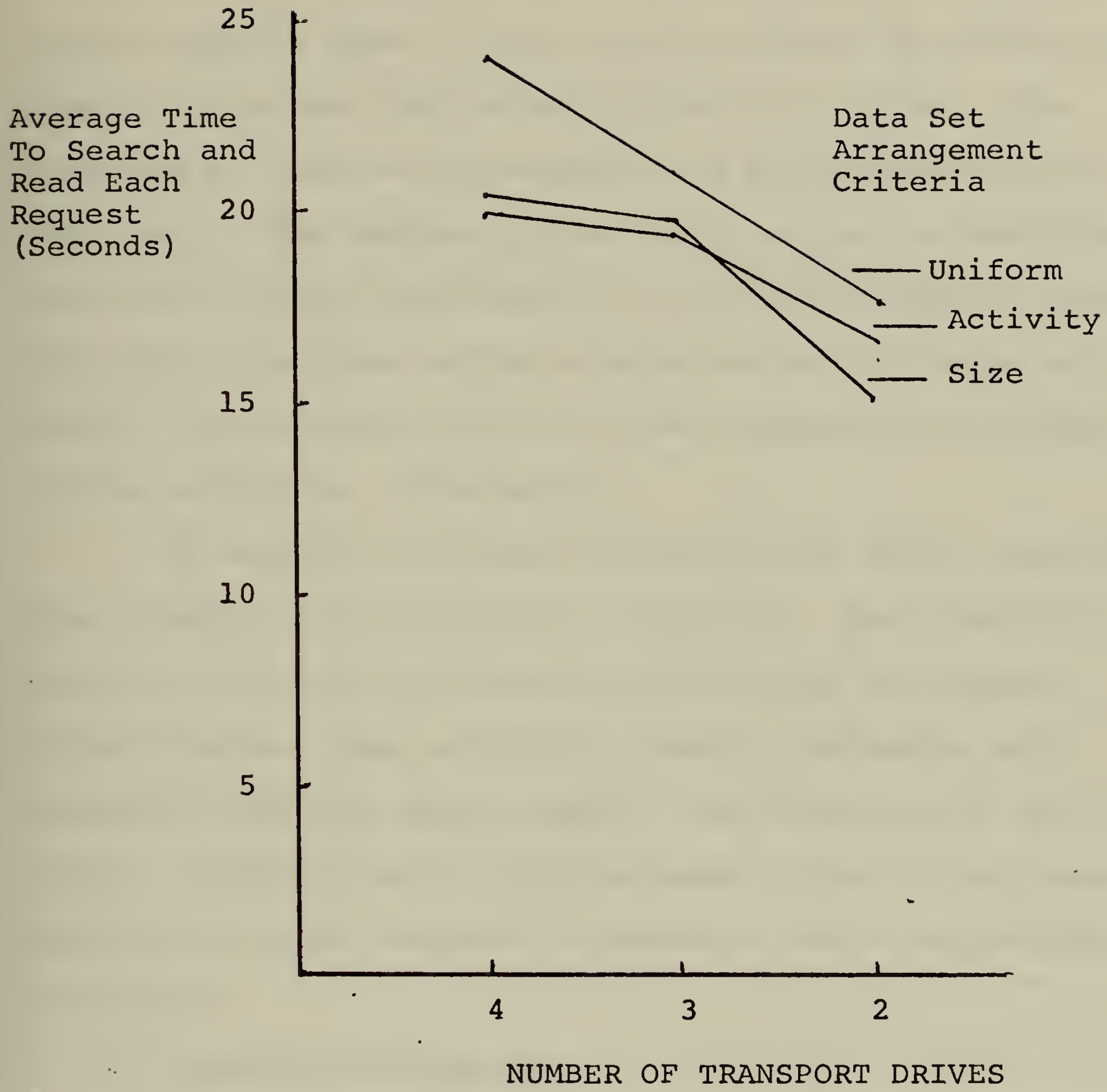


Figure 16 ARRANGEMENTS OF DATA SET ON TAPE

the center of a tape. In the proposed and tested system, the probability of a large data set having a high hit ratio is so small that the statistical effect to the average search time is insignificant. Consequently this combination arrangement was not simulated by the model.

5. Analysis

The arrangement of the data sets on tape by activity or size is an attempt to reduce the average search time. Figure 17 illustrates how average search time varies with the type of data set arrangement as well as the number of transport drives. When four transport drives are available and the delay in obtaining a drive is low, then a significant difference occurs between uniform arrangement of data sets and size or activity arrangements. As the number of transport drives decreases and the delay before obtaining one increase, the differences in search time decreases. The entire delay allows the queue to build for each topplate. The data set requests are arranged on the queue by increasing location on the tape. Consequently the average search time decreases as queues get longer until queue reaches length of 15-20. (See Figure 11.) An input rate of 90 jobs/hour puts a sufficiently heavy load on the model so that when constrained by only three transport drives, the advantage of data set arrangement is reduced, but not eliminated. The average search time reduced for all three arrangements as the queue length increased, but there is no assurance the queue length increased proportionally for all three



Constant: Minimum Queue Length 1
Input Rate 90 Jobs/Hour

The read time for each case is constant so only the search time varies.

Figure 17 AVERAGE TIME TO FIND AND READ EACH REQUEST

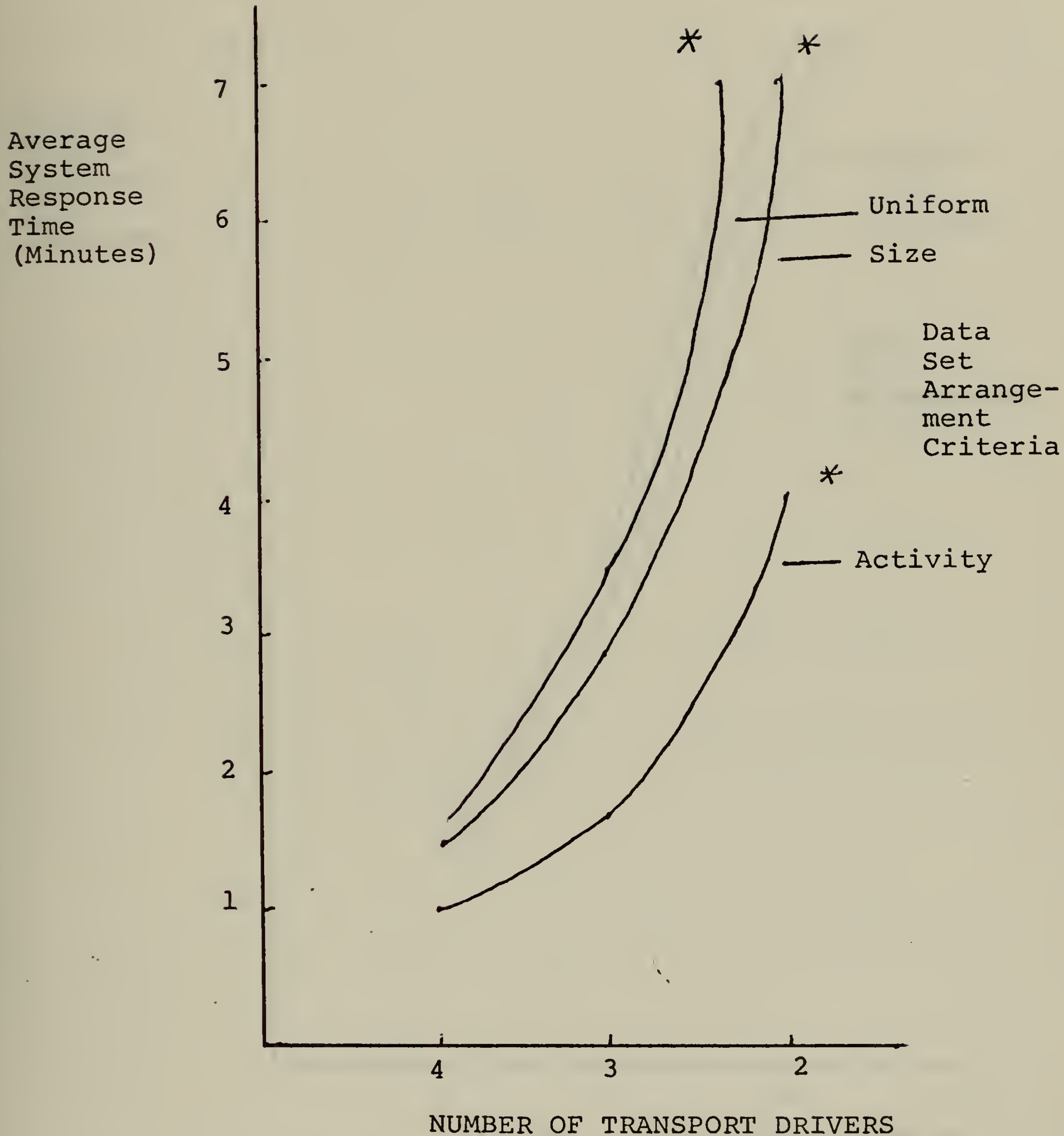
arrangements. The crossing of the activity and size curves is not considered significant. Since the search time did not give a clear answer the next best idea is to check the system response time. Figure 18 illustrates the average response time versus the number of transport drives. The advantage of data set arrangements is best illustrated in this graph. The average system response time is significantly lower for activity arrangements in all cases shown. Remember that these times are estimates obtained by simulation of a typical installation and they do not necessarily represent any one particular installation.

An analysis of system response time versus topplate queue length is illustrated in Figure 19. The results indicate that with four transport drives that the average system response time is almost linearly increasing with increasing topplate queue length. The linearity of the increase indicates most of the increase is due to the system waiting for enough topplate requests to arrive to initiate the search.

6. <u>Hardward Utilization</u>

The utilization of the entire TBM System hardware tends to depend on the utilization of the critical components of the system, the transport drive and the External Data Channel Processor. These two devices affect the usage of the other devices significantly.

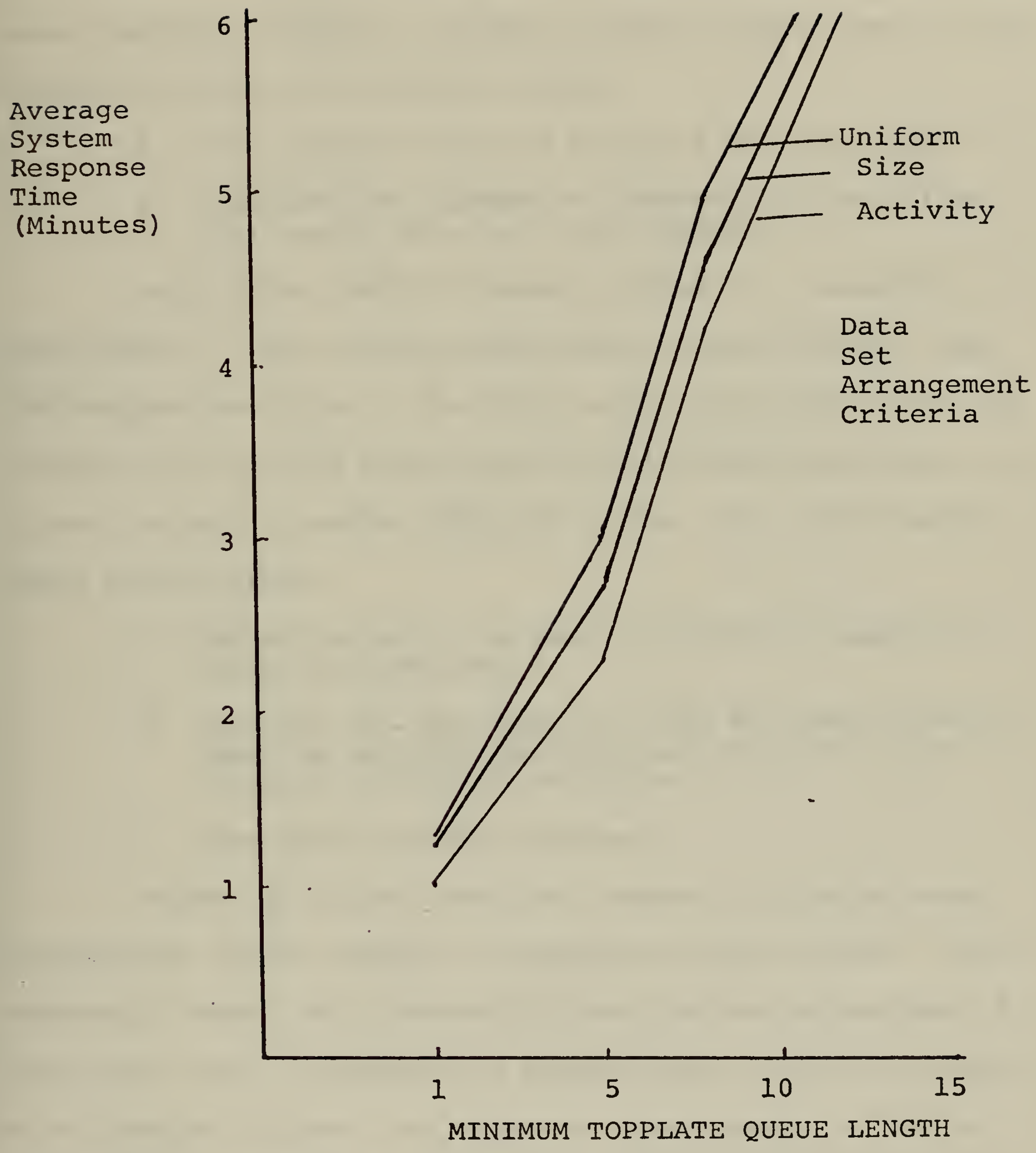
In the first situation where a large amount of data is to be read (or even worse, updated in place) without



Input Rate 90 Jobs/Hour

*Model Overflow - the time is an average of all transactions completed up until simulation stopped.

Figure 18 SYSTEM RESPONSE TIME VS. DATA SET ARRANGEMENT CRITERIA



Constant: Input Rate 90 Jobs/Hour
Transport Drivers 4

Figure 19 AVERAGE SYSTEM RESPONSE TIME VS. TOPPLATE QUEUE LENGTH

involving much search time, the result is a backlog for and delay due to the EDCP. An example of this would occur when six transport drives search short distances and then read a large number of blocks. If this is the problem that a user encounters, then he probably should:

1. Use another EDCP and increase his throughput or

2. Decrease the number of transport drives since they would have poor utilization.

More often the TBM* Memory System is transport drive bound. The average search time is much larger than the average read time. The EDCP may sit idle while all the transport drives are busy searching different topplates. If a user finds his system transport bound, then three major steps can be taken:

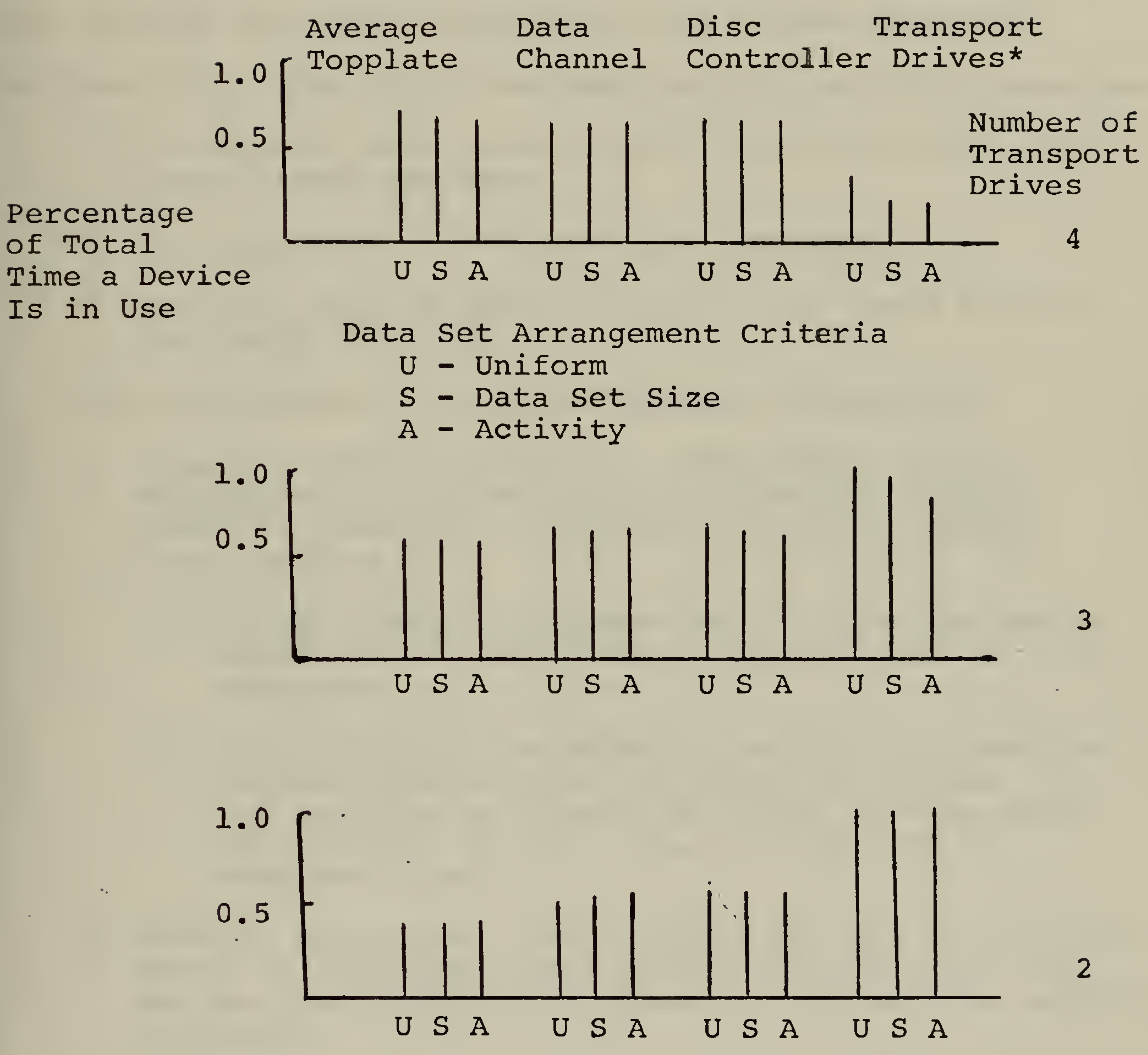
1. Queue requests for each topplate in sequential order by location.

2. Arrange his data sets on tape by some criteria, such as activity or data set size, in order to shorten average search time.

3. Use more transport drives.

Figure 20 illustrates the changes in the hardware utilization as the number of transport drives varies. The percentage shown for transport drives is the percentage of total time that a request was blocked due to non-availability of a transport drive. As requests were blocked at higher percentages of total time, then utilization of the topplates, the data channel, and the disc controller decreased.

INPUT RATE = 90 Jobs/Hour
MINIMUM TOPPLATE QUEUE LENGTH = 1



*Percentage of total time that a request was blocked due to non-availability of a transport drive.

Figure 20 HARDWARE UTILIZATION

V. CONCLUSIONS

The TBM* Memory System is one of the leading contenders in the recently developed mass storage technology. Mass storage systems offer many advantages over existing tape and disk systems and would be effective in a wide range of applications. Some major applications for the TBM System are:

1) To replace large conventional on-line and off-line tape library systems;

2) As supplements to enlarge disc systems;

3) To allow data to remain on-line that would usually be stored off-line.

The main features of the TBM* Memory System are:

1) Complete system redundancy. Any device of the system can be excluded by using another device like it, providing a duplicate is in the system configuration.

   a) In the case of equipment malfunction the whole system is not down waiting for one device to be repaired.

   b) It is possible to schedule maintenance downtime without sutting down the whole TBM System. If the downtime is scheduled during slow periods, then little, if any, delay is incurred in response time.

2) Modular Expansion. The installation can buy for its needs at the present and expand a little at a time as the installation increases in size and the activity increases.

3) Reliability. Due to equipment redundancy the system reliability is very good. Due to redundancy recording techniques the data reliability is also very good.

4) Large Library On-Line. The TBM* Memory System provides a feasible method of keeping a large data base on-line at all times.

5) Rapid response. A request to the TBM System can achieve varying response times depending upon the access techniques used. Due to a process of backfilling the host seldom sits idle waiting for the request to be answered. Backfilling is the process where the host puts a job on an inactive queue after sending data set request to the TBM* SCP.

6) Low Cost Per Hit. The large capacity of the TBM System gives the user an economical means of storing a large data base. Additionally the tape is reusable which reduces the costs.

Based on the results of the simulation model, the following conclusions are made:

1) The average system response time varies inversely with the number of transport drives and directly with the input rate.

2) One minute average response time is possible with: a) 2,3, or 4 transport drives if the arrival rate is 60 jobs per hour, b) 3 or 4 transport drives if the arrival rate is 90 jobs per hour, c) 4 transport drives if the arrival rate is 120 jobs per hour.

3) At the arrival rate of 90 jobs per hour the average system response time more than doubles for each transport drive less than four.

4) At the arrival rate of 60 jobs per hour the decrease in average system response time due to adding more than 2 transport drives to the configuration is less than 20 seconds. (Response decreases from 50 seconds to 30 seconds.)

5) The average time that each data set request spends waiting for transport drive increases as the number of transport drivers decreases and varies directly with the average system response time.

6) If the arrival rate is 60 jobs per hour or greater and the data is located on a minimum of 5 topplates, then more than one transport drive is required to prevent a backlog of jobs indefinitely.

7) At the arrival rate of 120 jobs per hour a minimum of 3 and preferably 4 transport drives are needed.

8) As the arrival rate increases from 90 to 120 jobs per hour with only three transport drives available, the average number of requests waiting on a topplate queue increases from less than 1 to over 3. At times the queues get very long. If 4 transport drives are available then the average queue length is less than 1 for all the arrival rates tested.

9) The number of transport drives has no effect on the average search time once the requests have been released from the queue to be searched.

10) The logical minimum topplate queue length was found to be between 10 and 15 requests. This minimizes the average search time per request, but not necessarily the average system response time. (See numbers 11 and 12).

11) The average system response time can actually be increased by setting the minimum topplate queue length between 10 and 15 due to the requests sitting in the queue for long periods waiting for enough requests to arrive to meet the required minimum.

12) If sufficient transport drives are available such that waiting for a transport drive seldom occurs then the average system response time increases linearly with increasing topplate queue length. The linearity indicates that jobs waited long periods in the topplate queue waiting for enough requests to arrive. Thus in the case of low input rates, the advantage of faster search times by queuing requests is defeated by having to wait to fill the queue.

13) Minimum topplate queue lengths make significant savings in system response time if many data set requests arrive in a burst or from high input rates.

14) If the minimum topplate queue length is 5 or less then 3 transport drives would reduce system response time significantly. For example, at 90 jobs per hour with a topplate queue length of one, the system response time is reduced from 6.5 minutes to 0.75 minutes when the number of transport drives is increased from 2 to 3. If the minimum topplate queue length is greater than 5 then the reduction in system response time from the third transport drive is less than one minute. The fourth drive makes no difference in system response time.

15) The average search time to locate a data set is dependent on the data set arrangement criteria. Size and activity arrangement produce savings of 5 to 15 percent when compared to uniform arrangement.

16) Data set arrangement criteria of size or activity produce significant savings of near 50 percent in average system response time. The amplification of savings is the result of queuing more data set requests before searching which leads to more efficient searching.

17) Arrangement by activity reduces average system response time more than size arrangement criteria.

18) At 90 jobs per hour with 4 transport drives the average utilization of the 5 topplates, the single data channel and a single disc controller were all about 70 percent. The transport utilization was 20 to 35 percent. With three transport drives, the average utilization of the transport drives increases to near 100 percent causing the utilization of the other devices to decrease to about 65 percent. With 2 transport drives the system was definitely transport drive bound with their utilization at 100 percent and the other device utilization decreasing to 40 to 60 percent.

19) Thus the utilization of the TBM* hardware is inversely proportional to the non-availability of transport drives. Higher utilization of hardware is possible by adding extra transport drives to the system configuration, (a maximum limit of six).

A. RECOMMENDATIONS

The author makes no claims to have tested all the possible configurations of the TBM* Memory System. The recommendations stated are based on the analysis derived from the model and the research performed.

The request loads tested in this research are 60, 90 and 120 jobs per hour resulting in data set requests of approximately 300, 450, 600 per hour. The following are equipment

recommendations with exceptions included. The exceptions are possible cases which may occur at certain installations.

1. Equipment Recommendations

The following recommendations are made assuming the user requires normal operational reliability. If the users require very high operational reliability then at least two of each hardware device should be included in the system configuration.

1) Only one supervisory control processor is required. Unless the installation could make use of a redundant SCP (PDP-11) in some other applications, then there is no need for a second SCP.

2) Only one External Data Channel Processor is required. The host's throughput requirements, as derived by the hit ratio of data within TBM blocks and the user's data manipulation capabilities, determine whether throughput should be greater than 0.75 megabytes/sec. Only if the user requires higher throughput should an extra EDCP be used.

3) Only one data channel is recommended. In read applications, the EDCP and the DC are busy at the same time, one connected to the other while the data flows through. Consequently only one DC would be necessary unless a lot of updating is to be performed. If a large percentage of referenced files are to be updated in place or written on a write tape, then a data channel will be tied up not only with the write data transfer but with the write verification. In the write case, the data channel is used more and may become a bottleneck. If having just one data channel does delay the system significantly, then another data channel should be added.

4) Two Disc Controllers. The disc storage is used by the TBM system for file directory information and data transfer to the host. The host uses the disc storage not only for data transfer to the TBM System but also as storage for frequently referenced files and perhaps as a work storage area. The only path to the disc storage is

through the disc controller. Only one entity can have a controller at once so if only one disc controller was available and the host was in control, then the TBM system would have to wait. The dual disc controller system allows one entity to have a disc controller and one disc drive, while another entity can use the other disc controller and reference any other disc drive. This method results in very few delays and each entity has the benefit of access to disc storage. The amount of disc space required is host dependent. The TBM memory system needs only a file directory on disc and sufficient room to store one TBM block (this, of course, would be a ridiculous waste of TBM capabilities).

5) Transport Drives. The TBM* Memory System will be transport drive bound in normal applications. The number of transport drives to include in the TBM System depends upon the desired system response time. The model can adopt to any configuration and the user should try to estimate his needs and then simulate the proposed system with the model. In general a user will find:

   a) If he has a lot of topplates and an equal hit ratio, then the minimum topplate queue length should be very low and an extra transport drive may be necessary.

   b) If he has a certain amount of data spread out over thirty topplates, then he will need more transport drives than if he had the same amount of data over fifteen topplates.

   c) If many requests enter the system at the same time, they should be batched at the topplate until all requests have been received, and then the search should begin.

   d) If a continuous high input rate exists, the minimum topplate queue level should be set between 10 - 15.

   e) If a continuous low input rate exists, then the minimum topplate queue level should be set to one.

6) Dual Transport Modules. The number of dual transport modules to include in the user configuration is totally dependent upon the amount of data the user wishes to have on-line. One Dual Transport Module is required for every

two topplates. One possible application of the TBM system is to store off-line information in a condensed form. One TBM tape is equivalent to 500 conventional tapes and can be stored in an off-line library in the space of a single 3-inch 3-ring binder. In this application only one Dual Transport Module is necessary since the tapes are taken off-line after being filled. Another application is for interactive CP storage. In order to minimize access time, the data is stored on one-seventh of a tape. In this application many more dual transport modules are necessary than normally used for the same amount of information.

2. Summary

The individual user must carefully define his needs before selecting a particular memory technology or memory system. The TBM* Memory System uses new mass storage technology and is indeed rather complicated. The system model is presented to help the user understand the system. The sensitivity analysis is presented as a guide for the user in selecting the proper configuration for his installation.

APPENDIX A

TBM* MEMORY SYSTEM HARDWARE

A. COMMUNICATION AND CONTROL SECTION (CCS)

The control section consists of the System Control Processor (SCP), which is a PDP-11 computer, External Data Channel Processor (EDCP), discs, and channel switching hardware.

1. Supervisory Control Processor

The SCP controls overall functioning of the TBM system. The SCP interprets requests for search and data transfer operations initiated by the host computer. The SCP then allocates Transport Modules, Transport Drivers, Data Channels and disc space necessary to complete the operation. A basic description of the PDP-11 by digital corporation is given in Figure 21.

2. External Data Channel Processor (EDCP)

The External Data Channel Processor is usually a PDP-11 like the SCP. The purpose of the EDCP is to act as an interface between the TBM* Memory System and the shared discs. The throughput of the TBM System is limited by the number of EDCP's. Reasons for having an EDCP instead of directly transferring data from the data channel to the disc are complex. A brief explanation of the reasons are as follows:

a. The transfer rate of the TBM* Memory System is approximately 6 X 10^6 bits per second on either

Figure 21

CHARACTERIZATIONS OF THE PDP-11

Manufacturers & Model	PDP-11 by Digital[1]
DATA Formats	
Word length, bits	16
Fixed-point operand length, bits	16
Instruction length, bits	16/32/48
MAIN STORAGE	
Storage type	Core
Cycle time, microseconds/word	1.2
Minimum capacity, words	4,096
Maximum capacity, words	124K
Parity checking	Optional
Storage protection	No
CENTRAL PROCESSOR	
No. of accumulators	8
No. of index registers	Up to 8
No. of directly addressable words	32, 768
Indirect addressing	One-level
Add time, microseconds (full word)	2.3
Hardware floating point	No
Hardware byte manipulation	Standard
Immediate (literal) instructions	Standard
Power failure protection	Standard
Real-time clock or timer	Optional
INPUT/OUTPUT CONTROL	
I/O word size, bits	16
Direct memory access channel	Standard
Maximum I/O rate, words/sec	833,000
No. of external interrupt levels	Variable
PERIPHERAL EQUIPMENT	
Disc pack storage	Yes
Non-interchangeable disc storage	Yes
Drum storage	No
Magnetic tape speed, cps	No
Punched card input speed, cpm	200
High-speed paper tape input, cps	300
High-speed paper tape output, cps	300
Other standard peripheral units	50 DECtape, CRT displays, A/D converters, printers, etc.

[1] Datapro 40 Features and Surveys, "All About Mini Computers," Datapro Research Corporation, 1972.

Manufacturers & Model	PDP-11 by Digital
SOFTWARE	
Assembler	2-pass
FORTRAN compiler	Yes
Other compilers	BASIC
Operating system	Yes
AVAILABILITY	
Date of first delivery	March 1970
Number installed to date	Over 1000

the read or write channel of any data channel. The transfer rate of the 3330 disc is approximately 6.4 X 10^6 bits per second. Neither system can be slowed down or speeded up. The difference in transfer rates would be disastrous if an interface buffer was not used.

b. Originally the host put the data sets on the disc and entered the address in its disc file directory. These data sets were then stored in one or more TBM blocks. The host then signals the TBM* Memory System that a particular data set is wanted and to put the data set in just a certain location that is now available on disc. The host is expecting that data set to be in that location and has notified its file directory in order to get to the data set. The data stored on TBM tape is addressed for the original location so the address must be changed. This address changing is done on the fly by the channel simulator just before the data goes to the disc controller.

The PDP-11 used as the EDCP must contain at least 3 core buffers of 16 K bytes each in order to allow the TBM* memory system to send data continuously. The fourth 16 K byte buffer is an extra which, when used, allows the TBM* Memory System to send data continuously and the 3338 disc to accept data continuously or vise versa, (see Figure 22).

Since one TBM block is approximately 130 K bytes and any number of blocks can be read as long as disc space is available, then it is obvious that the data cannot stop in the EDCP. The 16 K byte buffers are used in a flip-flop manner with the data channel filling several buffers before data starts transferring to disc.

3. IBM 3330 Disc System

According to Ampex, the best disc system to be connected to the TBM* Memory System is the IBM 3330 disc system. The 3330 disc system has a transfer rate close to

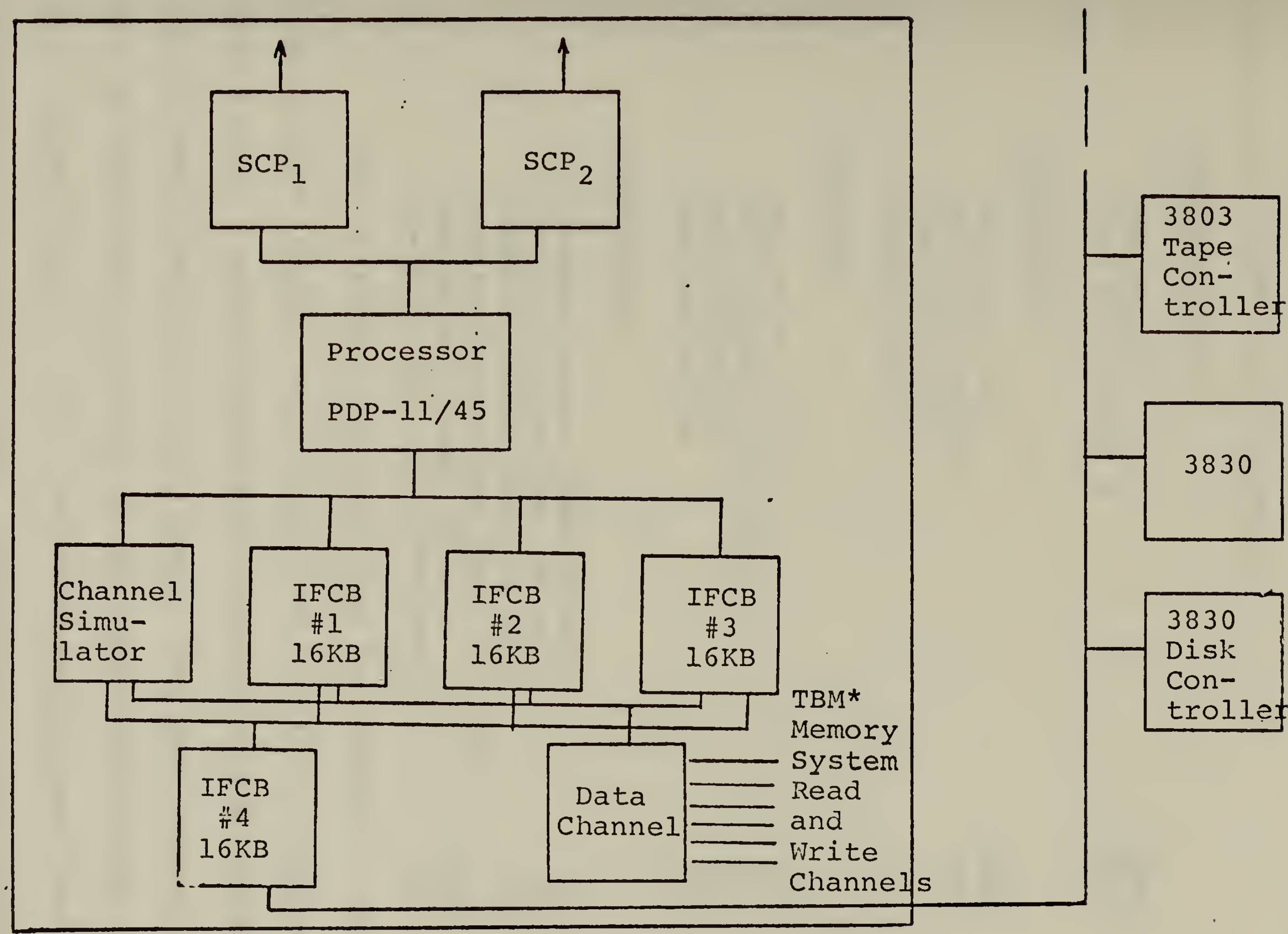


Figure 22 EDCP BLOCK DIAGRAM

the TBM* transfer rate. The transfer rates are not the same, consequently the 3330 disc system is not optimal for the TBM* Memory System. The IBM 3330 disc drive with corresponding disc pack #3336 and disc controller #3830 is the most advanced of existing disc systems that satisfy the needs of the TBM* memory system.

<u>IDENTIFICATION</u>	
Disc Pac #	3336
Disc Controller #	3830
<u>TECHNOLOGY</u>	
Density - BPI	4040
No. of Cylinders	404
Removable Pack	
Movable Head	
<u>CAPACITY</u>	
Per Drive (M bytes)	100
Per Cylinder (K bytes)	247
Min/controller	200
Max/controller	800
<u>PERFORMANCE</u>	
Minimum Access Time (M sec)	10
Maximum Access Time (M sec)	55
Average Access Time (M sec)	30
Average Rotational Delay (M sec)	8.4
Rotational Speed (RPM)	3600
Transfer Rate (K bytes/sec)	806

One cylinder of a 3330 disc is capable of containing two TBM blocks. Each 3330 disc has 400 cylinders for user data. A quick computation shows that 800 TBM blocks can be contained on one disc pack. The TBM* Memory System has to format all its data in the large TBM blocks (i.e., 130 K bytes/block). The user would seldom need that much information at any one time. Consequently the methodology

assumes that user outputs data onto disc in small segments and keeps a disc file directory of where the data was placed. As far as the user knows, all of his data is assumed to be stored on disc.

The purpose of the TBM* Memory System is to gather all the segments and form 130 K byte blocks. These blocks are then written on TBM tape and the TBM file directory knows all the data sets located within each TBM block. Whenever a user wants a data set stored on TBM tape, the host computer lets the SCP know what data set and where to put the entire TBM block on disc. When TBM has put the data on disc, then the SCP signals the host that the data is available and the user's data set, perhaps a small portion of the TBM block, is read from disc as though the data has always been on disc.

B. DATA STORAGE SECTION (DSS)

The data storage section consists of:

A. 1 to 6 transport drivers.

B. 1 to 32 transport modules, each consisting of two topplates.

C. 1 to 3 data channels.

Figure 23 shows an expanded data storage section. As illustrated in the figure, data never passes through the transport drivers, only through the data channel and to the transport module. The switching matrices allow any of the six transport drivers to be connected to any of the 64 topplates. Once this is done and the search has been completed,

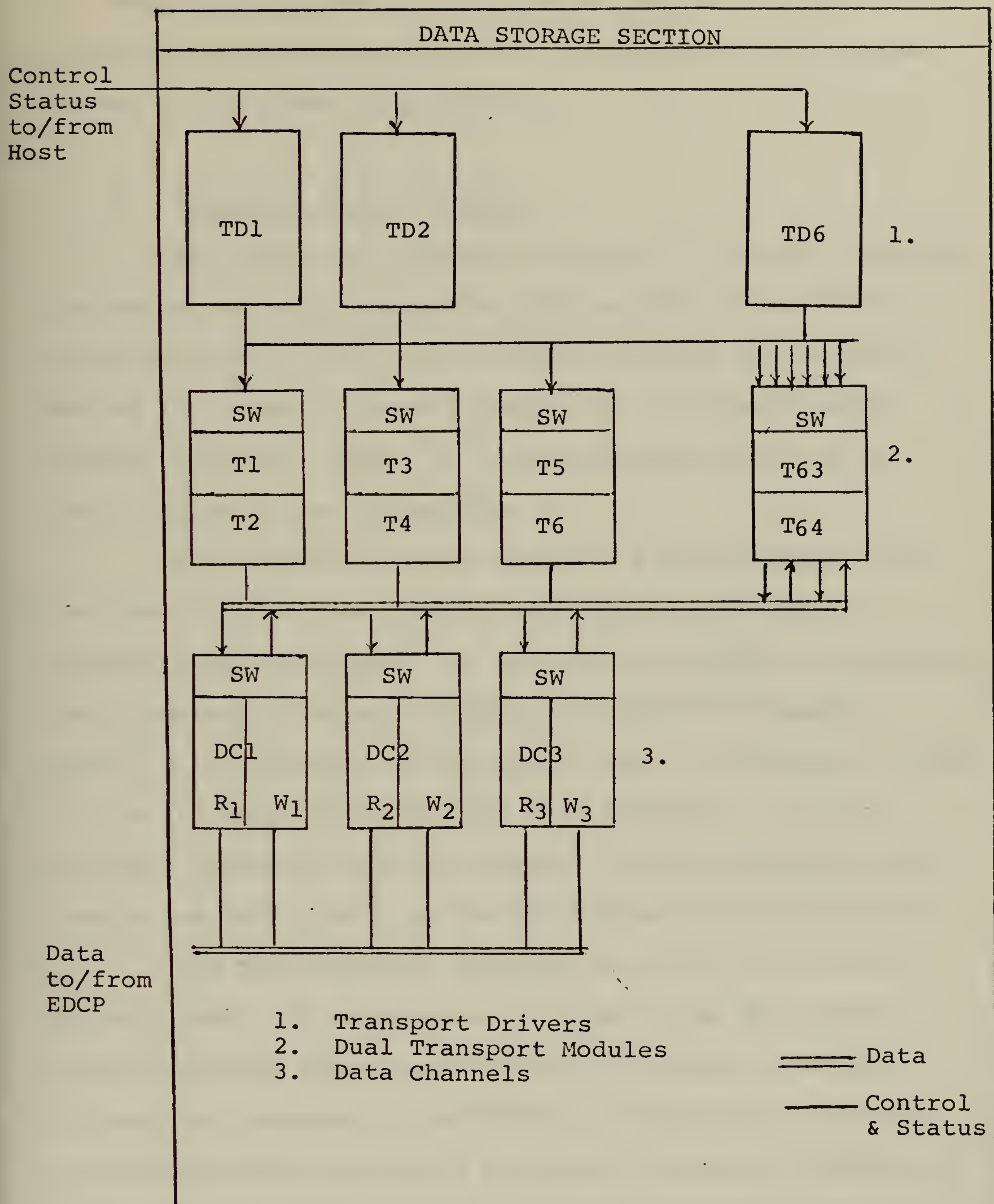


Figure 23 EXPANDED DSS CONFIGURATION

the resulting combination of transport driver and topplate can be connected to either the read or write channel of any of the three data channels.

1. Transport Driver Module

The number of Transport Drivers in a system determines the number of tape transports which can be in operations simultaneously. Up to six Transport Drivers can be implemented in a single system allowing for six simultaneous address searches (seeks) or read/write operations on as many different tape transports.

The transport driver contains a mini-computer and the power, servo, and control electronics necessary to operate a tape topplate. As directed by the TBM System Control Processor, the mini-computer controls the topplate to which it is connected during search and, in addition, a read or a write data channel during data transfer. It also monitors transport and data channel status and reports I/O completion with status to the TBM System Control Processor.

The mini-computer operates in a real time process control mode. It operates as a slave to the TBM System Control Process receiving high level operation commands. It executes commands by performing all the detail control and communication functions required by servos, address and tally track functions as well as read/write data channels. The mini-CPU is used extensively for system diagnostics, alignment and maintenance.

(This page intentionally blank)

The System Control Processor continuously monitors the status of the Transport Driver CPU. It also has the capability of reloading the mini's core memory as required without manual intervention. (See Figure 25).

User data is handled exclusively via the TBM Data Channel Module. The Transport Driver involvement in data transfer is limited to channel control and error reporting to the System Control Processer.

2. Dual Transport Module

The number of dual transport modules determines the overall storage capacity of a TBM system. A single system may have a minimum of one and a maximum of 32 dual transport modules, each with a capacity of almost 10^{11} bits.

A dual transport module contains two tape topplates and the switching elements necessary to connect the tape transports to transport drivers. The switching elements are packaged in plug-in drawer units and can provide for both topplates to be simultaneously connected to any two out of a maximum of six transport drivers.

The tape topplates (see Figure 26) have all the mechanical elements necessary to move tape but only a bare minimum of electronic components. The electronics necessary to drive reel, capstan and video head motors, address and tally track read/write electronics, general control logic, etc., are located in the Transport Driver Module while the electronics required to read and write digital data on tape is in the Data Channel Module. This approach minimizes the

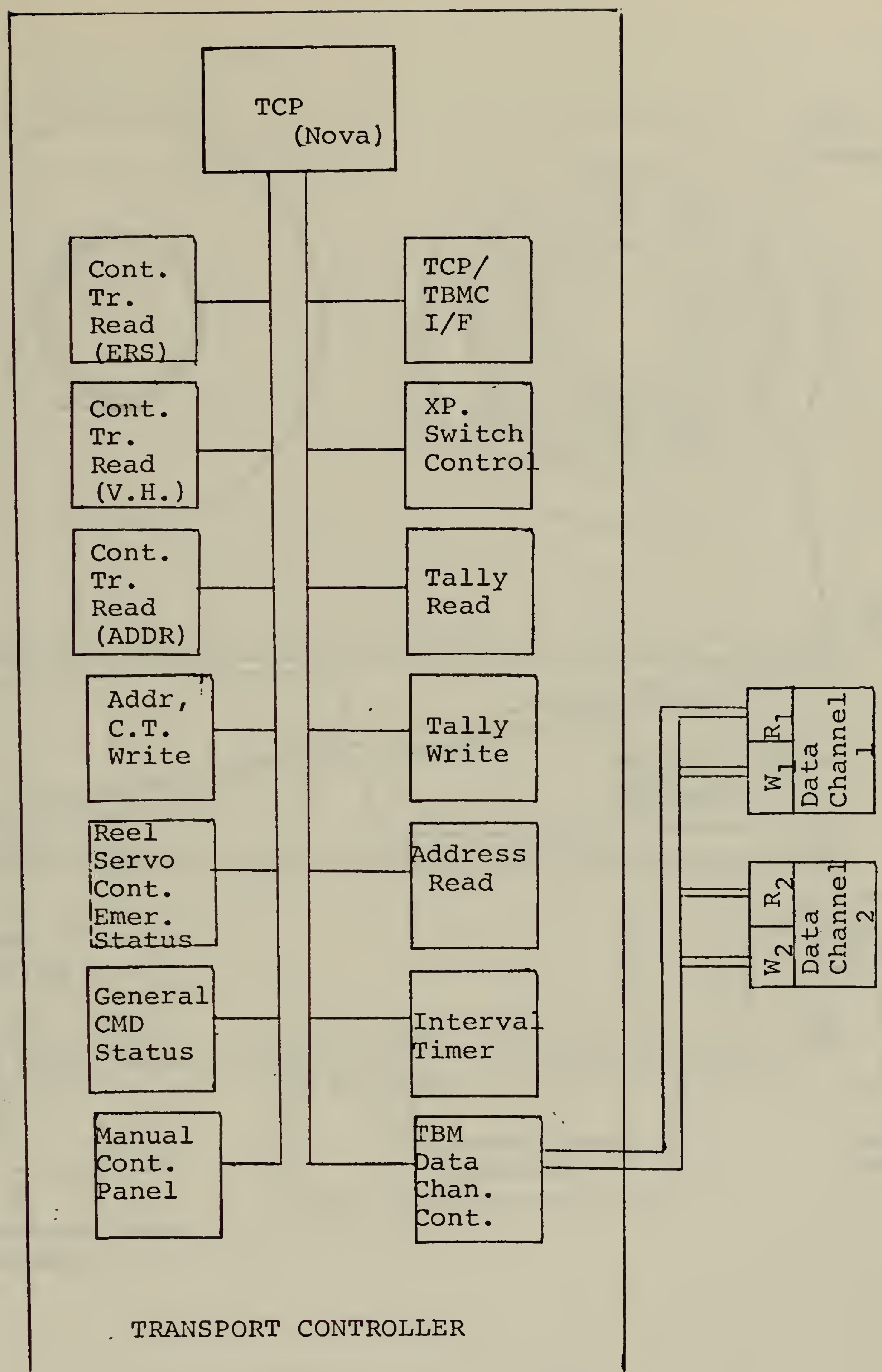


Figure 25 CONTROLLER AS A COMPUTER WITHPERIPHERALS

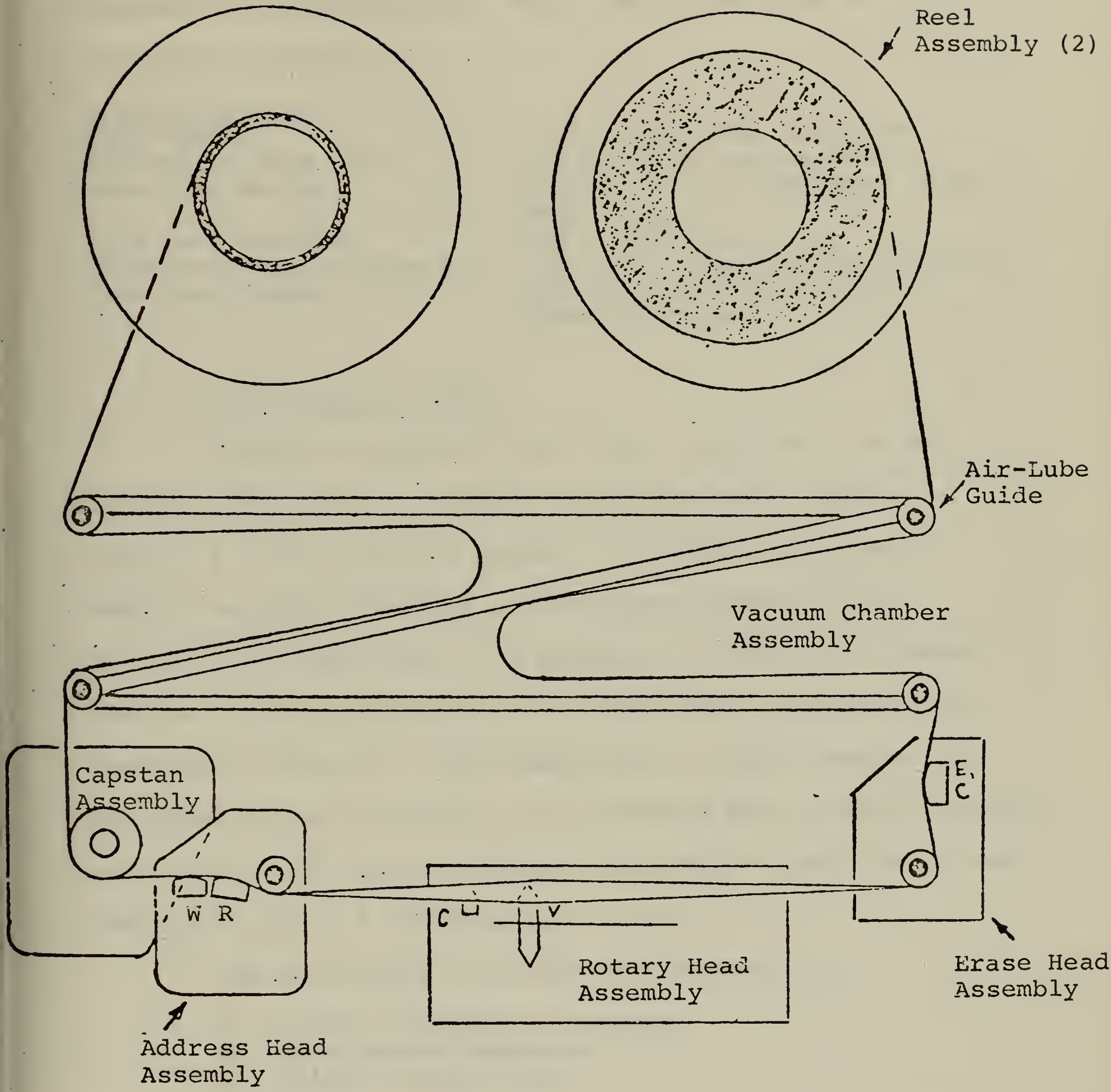


Figure 26 TOPPLATE

cost of expanding the on-line data capacity of a system because of the significant reduction in the cost of dual transport modules.

<u>Specifications</u>

Tape Speeds	:	5, 83, 248, 500, 1000 ips
Read/Write Data Rates	:	6×10^6 bits per second
Recording Media	:	Standard 2 inch magnetic video tape
Tape Reel Capacity	:	4.6×10^{10} bits
Transport Module Capacity	:	9.2×10^{10} bits (13 IBM 3330's)
Tape Data Format	:	Addressable Blocks (Std. block 10^6 bits)

3. <u>Data Channel Module</u>

Each Data Channel Module has a read and a write channel which can be operated simultaneously, each at a rate of 6×10^6 bits per second. A single Data Channel Module therefore provides a throughput capability of 12×10^6 bits per second. A maximum of three Data Channel Modules can be implemented in a TBM system, providing six read/write channels. Any combination of six channels can be operated simultaneously. The maximum Data Channel throughput is 36×10^6 bits per second, but EDCP's limit the system throughput to 12×10^6 bits per second.

The advantages of extra data channels are:

1) System redundancy (backup)
2) Maintenance downtime
3) Write verification

The Data Channel contains all the electronics necessary to handle the transfer of user data. In the write mode, data first enters the data interface buffer section. From there data is routed through the error detection and

correction encoding logic, through FM encoding electronics and is recorded on tape.

In the read mode, user data is reproduced from tape and restored by the FM read electronics, checked by the error detection and correction decoding logic, and then outputted via the read channel data interface buffer section.

The error detection and correction logic is implemented to correct any single bit error occurring within any 985 bit block of user data while detecting occurrences of larger error bursts. See Appendix B.

The Data Channel Module contains switching elements to simultaneously connect the read and the write channel to any two of a maximum of 64 tape transports. Figure 27 shows the Data Channel's functions as data is sent through the Data Channel.

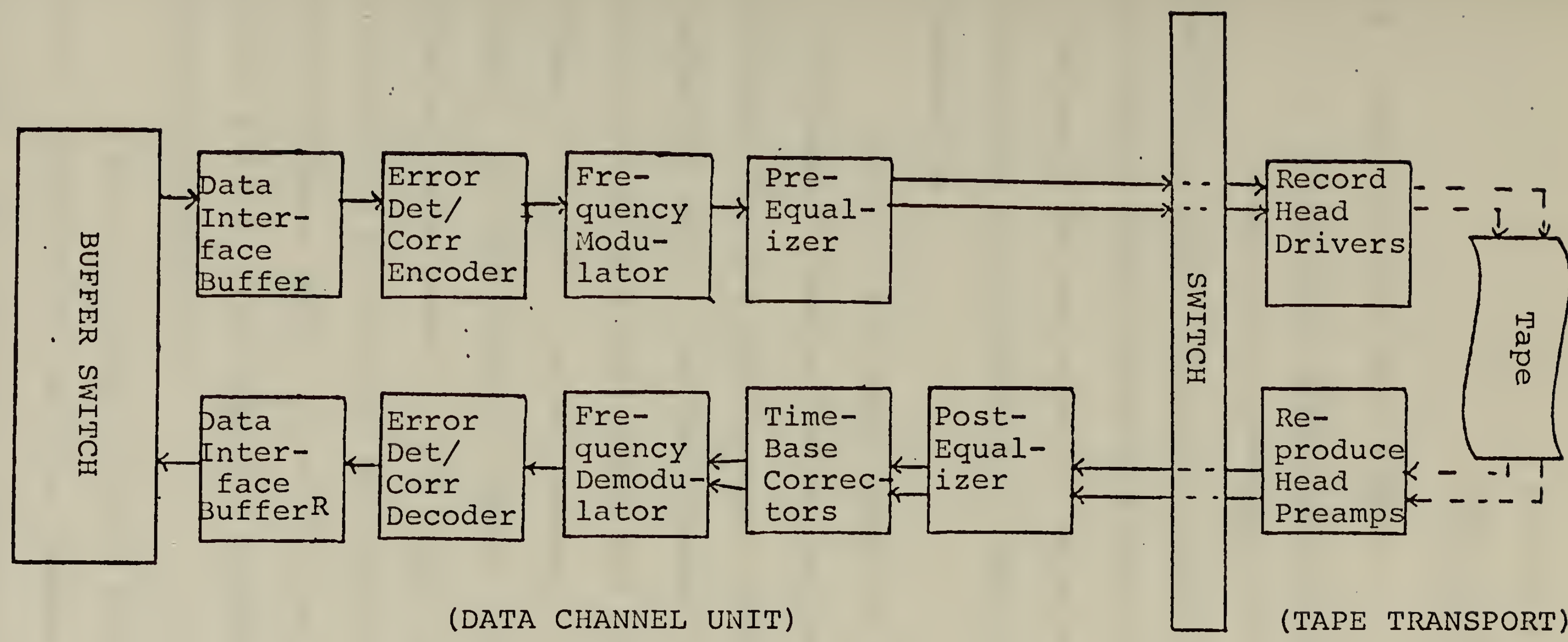


Figure 27 DATA CHANNEL SIGNAL FLOW (PRODUCTION)

APPENDIX B

RELIABILITY

A. DATA TRANSFER RELIABILITY

1. Data Error Detection and Correction (EDC)

Every bit is recorded twice on two adjacent video vertical tracks. They are separated by a vertical distance of approximately three-fourths inch. Errors due to defective spots on new tapes or defects caused by usage (i.e., scratches) are reduced and nearly eliminated by the separation.

Two of the eight heads on the rotating head drum are always in contact with the tape, (see Figure 28). This provides for the simultaneous recording and reading of the redundant data. The data channel takes the redundant data from each of the read heads and simply combines the signals. Dropout in either signal causes a drop in signal level rather than a complete loss of signal. Thus no decision is required when an error occurs. The logic implemented in the channel will detect and correct any single bit error in a 985 bit block of user data. It will detect occurrences of larger error bursts. Re-reads are automatically initiated if error bursts are detected.

The following is how a TBM block is broken down for error detection by the data channel as the data passes through. One TBM block consists of 182 segments or 1091 code words. Therefore, each segment contains six code words

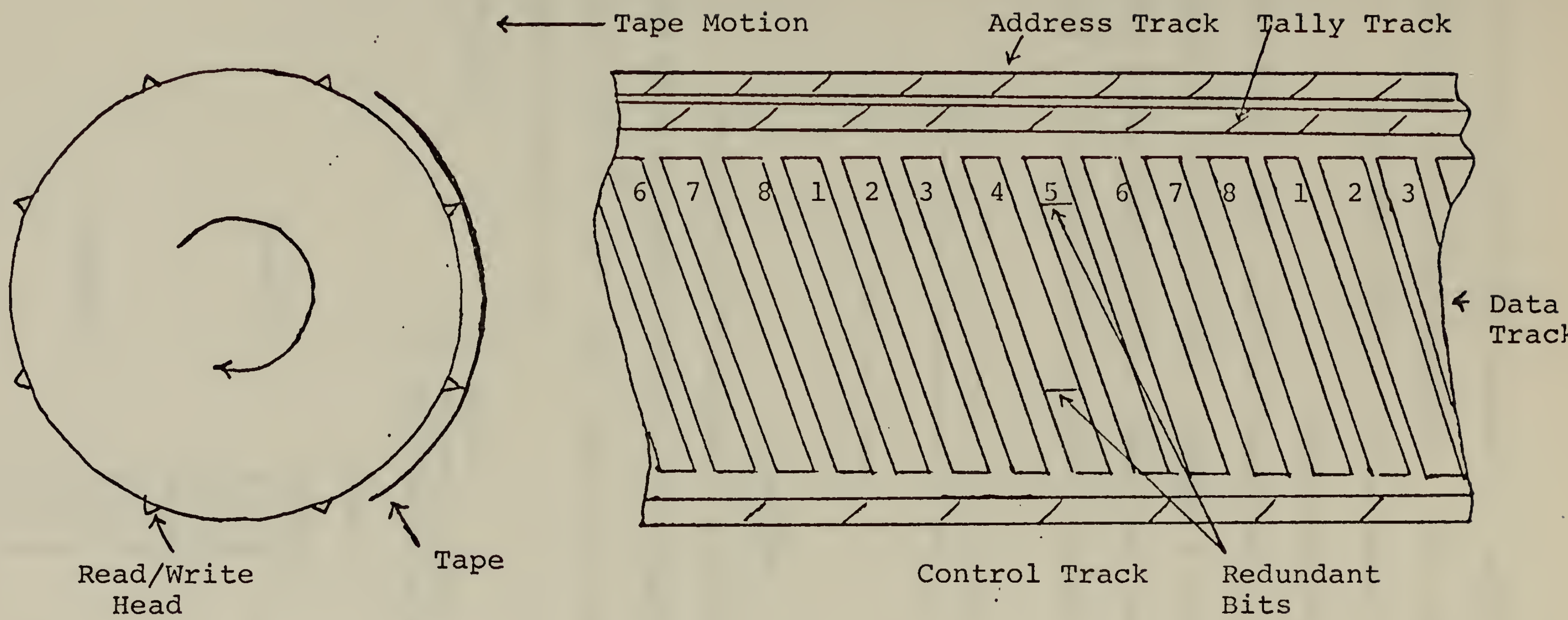


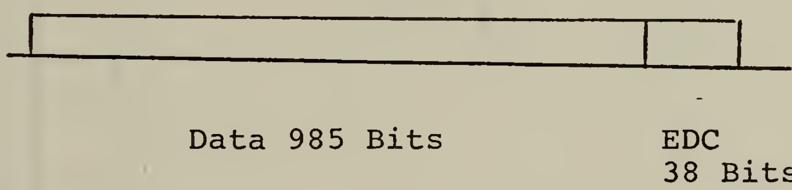
Figure 28 REDUNDANT RECORDING

except for the first segment of each TBM block which contains only five segments. (See Figure 29).

One TBM block = 182 segments

= 1091 code words

Therefore, each segment equals six code words (last 181 segments of each block), except for the first segment which equals five code words. Each code word consists of 985 bits of user information and 38 error detection check bits.

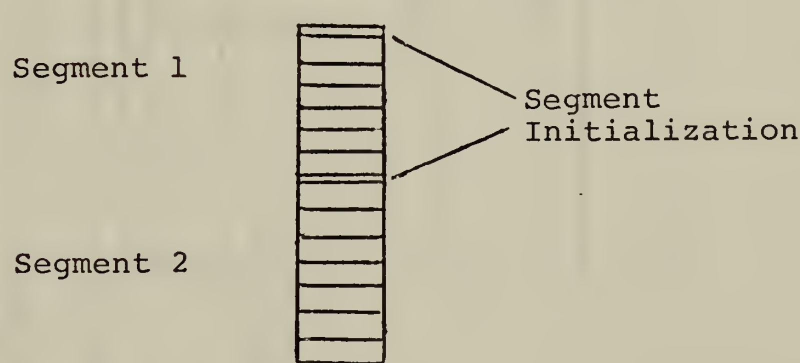


CODEWORD

There are two segments on each track before the first code of each segment which is a segment initialization. At data transfer rate read speed of 5 ips, the segment initialization takes 32 m sec.

Example: 1 track:
2 segment/track:
6 code words/segment:

Track A



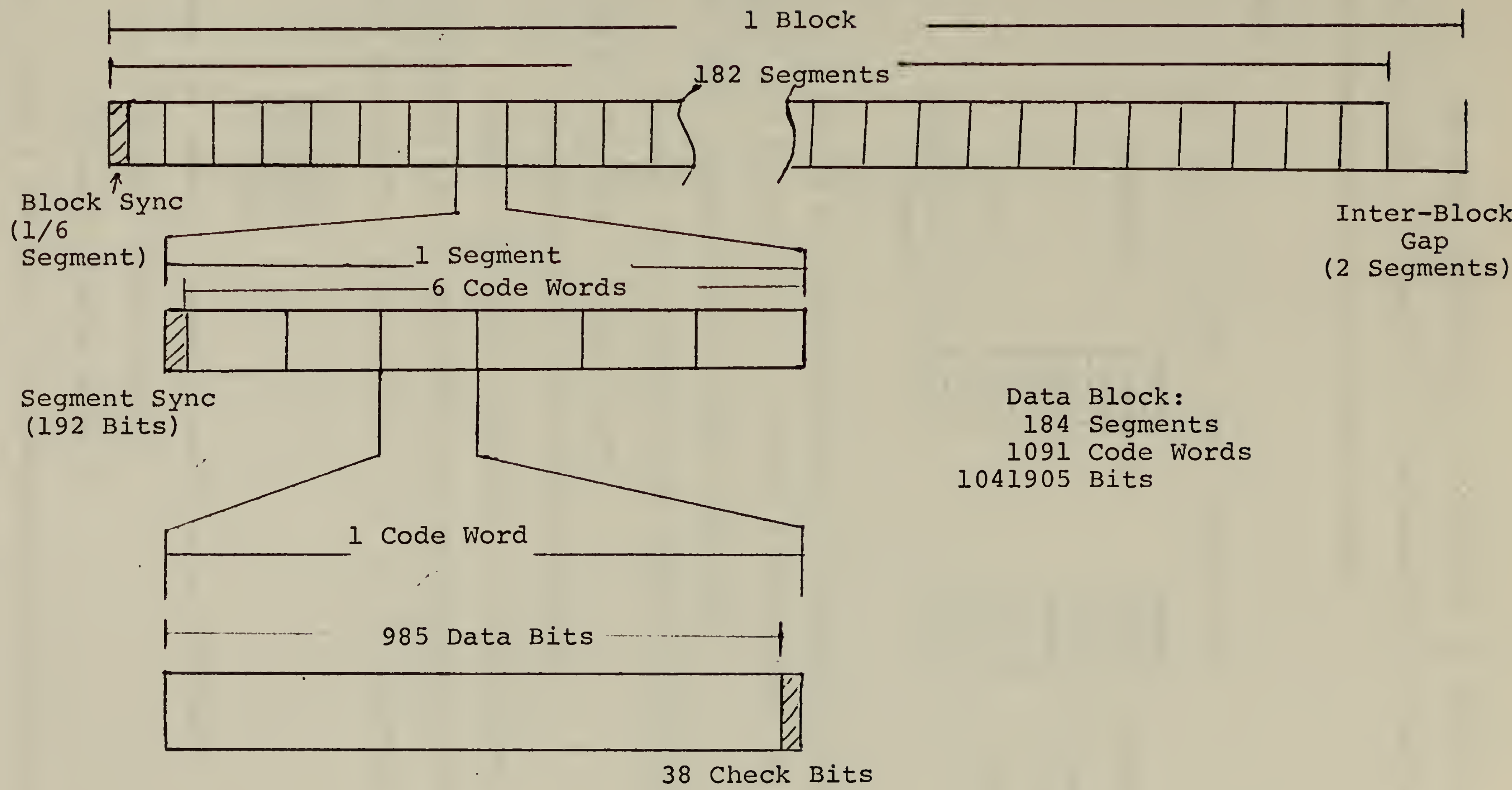


Figure 29 DATA BLOCK ORGANIZATION

TBM uses redundancy of recording in error detection and correction so each bit of information is recorded twice , simultaneously, by two different heads.

Example:

Two Tracks: Track 1000 & Track 1001

The 38 EDC check bits are for detecting errors both single and multiple errors within a code word. An example will best explain how the check bits work.

Example of a single error detection and correction: A string of data is transferred.

10011101/10000100/10110011/10110011/

Suppose one error occurred in transmission. Suppose a 1 instead of a 0 for first bit transmitted. Logically stack each successive 8 bit string on top of the preceding one.

10011101
10000100
10110011
10110011

Count the number of ones in each row and column:

If odd: Denote with 1

If even: Denote with 0

(i.e., Row 1 has five 1's. Therefore the checkbit for Row 1 is 1)

	Check Bits for Rows
10011101	1
10000100	0
10110011	1
10110011	1

00011001 - Check Bits for Columns

Now by combining check bits for rows and columns, the result is a parity check bit word:

1001 00011001

Row Column

This is the resultant check bit word that is compared with the initial check bit word that was derived when the data was stored. If the two check bit words match, then data is assumed to be correct. If the check bit words do not match then a trace is performed to find out where the error occurred.

Example: First digit of the check bit words should have been a 0 instead of a 1. The data has one bit wrong, but the check word bit has two bits wrong.

Correct Data	Check Bit for Rows	Data after Transmission	Check Bits
00011101	0	10011101	1
10000100	0	10000100	0
10110011	1	10110011	1
10110011	1	10110011	1

10011001 Check bits for Columns 00011001

001110011001 Check bit code words 101100011001

Since the check bit words do not match, then a mask test is performed to find out what row and column do not agree.

Check Bit Words:

Before Storage	001110011001
After Storage	101100011001

Logically Compare (Absolute Difference):

Disagreement	100010000000
	Row Column

This indicates a single error in Row 1, Column 1. Now a comparison of check bit words is performed after switching data in Row 1, Column 1. If check bit word now agrees, then data is accepted. If check bit words do not compare, then some other error correction routine must be taken.

Example of multiple error: Suppose the first two bits of the same data were in error.

Correct Data	Check Bit for Rows	Data after Transmission	Check Bits
01011101	1	10011101	1
10000100	0	10000100	0
10110011	1	10110011	1
10110011	1	10110011	1
11011001	Check Bits for Columns	00011001	
101111011001	Check bit code words	101100011001	
Row Column		Row Column	

The logical comparison shows a difference of two bits again but both differences occur in the column part of the check bit code word. This signals user that a multiple error has occurred. In some cases, correction of a multiple error is possible by examining the code words, but generally this is not a good practice.

Example of another multiple error: Suppose the first bit of each byte was reversed.

Correct Data	Check Bits	Data after Transmission	Check Bits
01011101	1	11011101	1
10000100	0	00110011	0
10110011	1	00110011	0
10110011	1	00110011	0
01011001	Check Bits for Column	11011001	
101111011001	Check code words	010011011001	
Row Column		Row Column	

In this example there are four differences, all contained in the row part of the code word.

Example of where this system does not work: Suppose the first two bits of the first and third bytes are reversed during transmission.

Correct Data	Check Bits	Data after Transmission	Check Bits
01011101	1	10011101	1
10000100	0	10000100	0
10110011	1	01110011	1
10110011	1	10110011	1
11011001	Check bits for columns	11011001	
101111011001	Check bit code words	101111011001	

The data is different but the code words are the same. This demonstrates the complexity that is involved in developing a good parity check.

The TBM* Memory System is designed to detect and correct all single errors as well as detecting and initiating a re-read for all multiple errors. The goal was to achieve an uncorrectable error rate of no more than 1 in 2 X 10^{10} bits.

2. Simultaneous Copy Generation

The TBM* Memory System provides the capability of simultaneously writing data to multiple transports. This feature allows a user to create a backup tape or perhaps an image dump file while outputting his data to the active TBM file tapes.

While files or records put into TBM are normally distributed between several on-line TBM tapes, the same data can be sequentially recorded on a tape dedicated as an image dump tape. Data written into TBM appears sequentially on the dump tape. This requires that one topplate be dedicated for output. If the output exceeds the capacity of a single dump tape, it will be necessary to take the dump tape and place it in an off-line library. If data cannot be recovered from an on-line tape, the appropriate dump tape is retrieved from the off-line library, mounted and the appropriate block(s) read. An index file is generated for purposes of cross-referencing.

An example of this would be a large insurance company that has a large number of customers and several different daily updating routines. All data is stored on a conventional tape library system. The hit ratio will be small for each account, i.e., probably reference each account about twice a month for updating payments, finding accounts receivable, statistical information collection, etc. Normal procedure is to make a tape of the day's transactions in numerical sequence by account number. A portion of the day's transactions are read in with the corresponding accounts.

After updating the account, the result is written over the original or written on another tape. This means an abundance of tapes, tape drives, operators to load tapes, and also an extensive filing system for on-line tapes and off-line tapes. If an error occurs, such as a lost tape or wrong computations, then past daily transaction tapes can be used to work backwards to get previous status or the old accounts status tape can be used if retained.

The same insurance company could use TBM* Memory System and have daily old records saved either on-line or off-line. One TBM tape off-line is equivalent to 500 packed computer tapes at 800 B.P.I. and occupies the space of a three inch binder notebook. Daily transactions could be kept on one TBM tape, recently updated files on another tape, and past status of all changed accounts on a third tape. In addition, as the company grew, the TBM system could be expanded to keep pace with the company. This is a very basic idea of a practical application without elaboration to demonstrate the efficiency of the TBM* Memory System and its ability to store several general generations of data for security.

3. <u>Internal Copy Generation</u>

Duplication of data is quite often employed as a means of insuring that the data is not lost. The TBM* Memory System can do tape duplicating independently of the host computer. This is especially convenient when the host is busy with internal matters or undergoing maintenance. The

compactness of the data allows the TBM* Memory System to store much more data in much less space, both on-line and off-line.

Single blocks, logical files or whole tapes may be copied from a tape at any address to any other tape at the same or a different address. This procedure utilizes read/write loop connection hardware found in the TBM Data Channel Modules. This facility provides the user an ideal method for file packing, file backup (store off-line), and file organization optimization.

B. HARDWARE LIFE AND RELIABILITY

TBM read/write heads have an average life of 100 contact hours (range 80-120). Since there is no contact during searching operations, contact hours are accumulated only during read/write operations. A worn out head is refurbished at a cost of about $1500. Because of the high data rate, about 1.5 billion bits of data can be transferred with the 100 contact hours.

Any transport can be moved by any controller and data can flow through any channel and EDCP. Thus there is mulipath access to data. If a hardware component goes down or requires maintenance, TBM can re-configure to assure a continuous flow of data. Tapes can also be mounted on any drive.

System diagnostics and maintenance are supplied to run in the off-line mode. These programs may be run independently of the host CPU. In addition, a special programming

language is supplied for the off-line mode, which enables the operator to write his own test programs. The system diagnostic programs can isolate troubles down to the component level. Maintenance programs involve such jobs as positioning tape loops in vacuum chambers for best tape control, pre-addressing and pre-checking new tapes, and testing the operation of servos and interval timers. Experience has indicated that a given block may be read and written about 2000 times before significant tape wear occurs.

APPENDIX C

SYSTEM HARDWARE TIMES

A. THE DISCONNECT/CONNECT TIMES OF TRANSPORT DRIVER AND TOPPLATE

The SCP makes the decision to:

1) Stop the read head;
2) Disconnect the transport driver;
3) Connect T.D. to another topplate;
4) Lock the video head in place.

The decision was made very quickly. The slow part is in waiting for the electromechianical mechanisms to make the necessary movements. Many situations and combinations of these situations can occur which complicates the timing. The most important consideration is whether the tape head is in motion. If it is, then the head must be stopped before disconnecting. The disconnect/connect takes only 0.525 sec. But if the tape head must first be stopped, then a total of 0.9 seconds is necessary. If the tape must be searched, then the video head phase lock takes place during the search. But if the tape is already positioned when a connection is made, then an additional time of 0.9 seconds is added to the disconnect/connect time. In quick summary:

Disconnect/Connect Time:

Stop Head - Disconnect - Connect - VH Lock	1.9 s
Disconnect - Connect - VH Lock	1.1 s
Disconnect - Connect	0.525 s
Stop - Disconnect - Connect	0.9 s

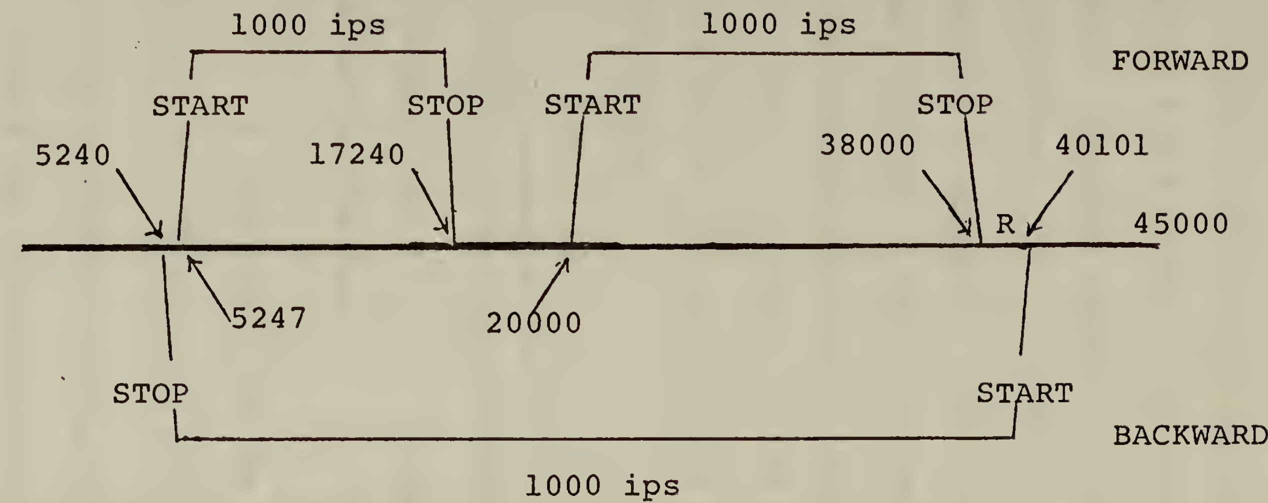
The times seem short until one realizes that over 1000 inches of tape could be searched in some situations.

B. SEARCH SPEEDS AND TIMES

The SCP makes the decision which topplate is to be searched next. If a transport driver is not connected to the topplate, then 0.9 seconds, on the average, is lost in making the connection. The transport driver knows what address it wants. After the connection is made the transport driver knows where the tape head is presently located and which way it has to go. A search can be made both forward and backwards. See Figure 30. The speed to which the transport driver accelerates is totally dependent upon the distance which has to be covered. The following table gives the figures.

Number of Blocks Greater than	Velocity Accelerated to	Time to Reach Velocity
200	1000 IPS	0.7 or 1.2 sec
46	248 IPS	360 msec
8	83 IPS	120 msec
0	5 IPS	15 msec

If the search is being conducted at any speed above 5 IPS, then a gradual slowdown through the speeds takes place. When the head is within eight blocks of desired address, then the minimum speed, 5 IPS, is used. The tally track for the chosen block is read and the read or write pre-positioning takes place.



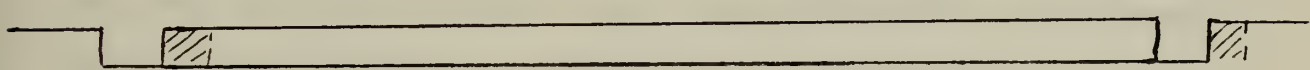
OPERATION	FROM BLOCK	TO BLOCK
SEARCH (F)	20000	38000
READ (F)	38000	40101
SEARCH (B)	40101	5240
READ (F)	5240	5247
SEARCH (F)	5247	17240
WAIT		

Figure 30 TAPE SEARCH

C. TIME TO READ ONE TBM BLOCK

A TBM block is equivalent to 1,041,905 bits or approximately 130 K bytes. One block uses 0.9752 inches of TBM tape. The data transfer read or write speed is 5 IPS. Time to read the entire block is 182 millaseconds (ms) or approximately 0.2 seconds. The resultant data transfer rate is approximately 6 mega bits/sec. The interblock gap, (I.B.G. - the distance between blocks) is 5 x 10 inches. At 5 IPS, the time to go over the IBG is 2 ms. The beginning of each block has an initialization area that contains block information

167 m sec 182 m sec 2 msec



$5x10^{-3}$ in 0.975 inch

IBG Initialization TBM Block IBG

A reread due to the data channel detecting a multiple error burst occurs only once out of every 3000 blocks read.

D. TIME TO WRITE ONE TBM BLOCK

The first consideration before writing is whether the tape has to be erased. Erasing is done at 5 or 83 ips dependent upon the number of blocks erased.

Less than 10 blocks	5 ips
Greater than 10 blocks	83 ips

If the tape is clear then the time to erase is not necessary. This shows one of many advantages of using a

write dump tape instead of immediate update (assuming the write dump tape has been pre-erased).

Advantages of using write dump tape:

1. No need to erase during busy period.

2. Update at a non-busy period.

3. Security backup tape.

4. Saves time (very important during busy period of day)

Disadvantages of using write dump tape:

1. Must keep a write file directory.

2. Records are not current.

The actual write is done at the data transfer read or write speed of 5 ips. Time to write is the same as the time to read, 182 millaseconds (ms). The resultant data transfer rate is approximately 6 megabits/sec. Each write is verified by re-reading just through the data channel's error detection section. This is also done at 5 ips. If no errors are detected then the data channel and disc space are released. Before the transport driver is released, the tally track is updated at 5 or 83 ips. The write is now finished and the transport driver is released. While the write has been transpiring, the SCP, in addition to controlling the write, has updated the write file directory if the system uses a write dump tape.

1.	Erase	5 or 83 ips
2.	Reposition	Normal search speed
3.	Write*	5 ips
4.	Reposition	Normal search speed
5.	Verify	5 ips
6.	Reposition	Normal search speed
7.	Update tally	5 or 83 ips

*Simultaneously the SCP updates the write file directory.

APPENDIX D

TBM TAPE FORMAT AND CHARACTERISTICS

User data is recorded on TBM tape in a block format. Each block contains one million (1,041,905) bits. The separation between blocks is 5×10^{-3} inches. Individual blocks can be updated in place, i.e., erased and rewritten.

There can be a maximum of 46,500 blocks per tape. Of these, a portion (less than a thousand) are dedicated to diagnostic and maintenance functions, while usually less than 0.5% are demarked as a result of permanent errors found during tape pre-testing. The demarcation indicator is located on the Tally track. A maximum of 45,000 blocks can be made available for user data.

Three longitudinal tracks are recorded on each tape, two of which are of interest to the user; the Address track and the Tally track, Figure 31. Associated with each block is a small segment of each track.

The Address track field associated with one data block contains both a unique tape number and a block address number. User tape block addresses are number sequentially from 1 through 45,000. Records or files may be accessed by physical addresses, i.e., tape number and block number. The user does not need to know on which transport a tape is mounted. The TBM System Control Processer maintains a table which relates tapes to transports.

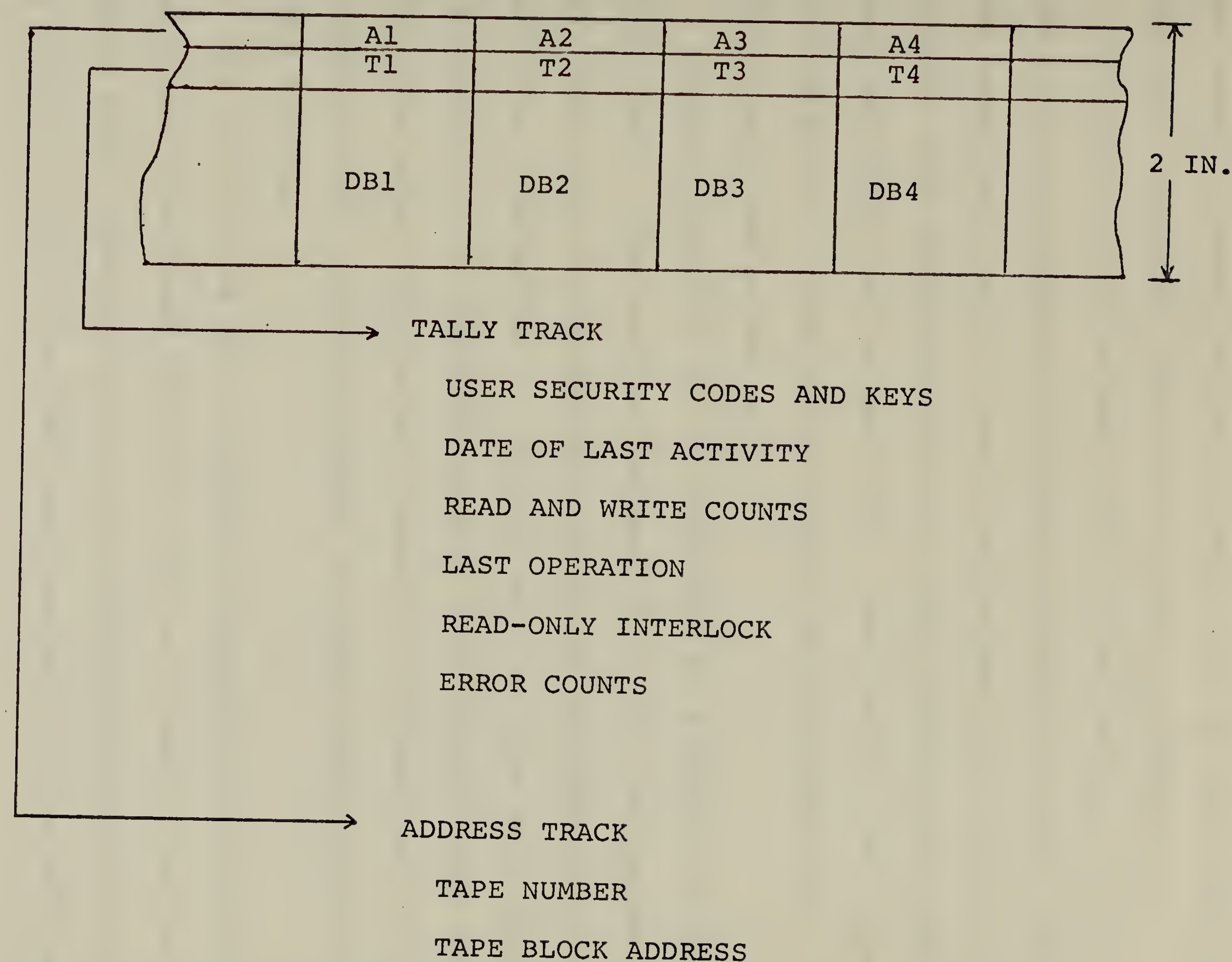


Figure 31 TAPE FORMAT

The Tally track is a user oriented facility. Each Tally track field associated with a data block contains user security codes and search keys, date of last activity, read only interlocks, and more as shown in Figure 32.

The Tally track is read prior to performing a block read, write, or erase, and updated after the process has been completed. At the user's option TBM will compare the security code words read from the Tally track to the code words supplied from the host CPU. Read, write, and erase processes can, at the user's option, be performed only if the Tally track code words match.

System statistics can be generated by reading the Tally tracks only. The user may request that only the content of the Tally track be transferred to it for purposes of generating system statistics. A typical usage of the Tally track information is to determine whether infrequently used files should be taken off-line and placed in a library or if those frequently used should be stored on faster devices such as discs or drums.

The Tally track can be updated independently from the data block associated with it. This is a useful feature if search keys, security codes or read only interlocks are to be changed. Tally track key searches can also be performed.

A. TAPE INFORMATION

Bits are recorded serially on the track at 7800 bits/ linear inch of track. (Note: tracks run almost horizontal across TBM tape as shown in Figure 33).

Figure 32 TALLY FIELDS

Byte	Bits	Function
1	Ø-5	Last Operation Codes (Octal):

Last OP	OK for Production	Demarked	Maintenance Permanent	Maintenance Updatable
Erase	25	26	- -	1
Write	45	46	15	5
Read	51	52	15	5

Byte	Bits	Function
1	6	
2	Ø-6	22-bit User Identification Number (IDNUM)
3	Ø-6	
4	Ø-6	
5	Ø-3	Month of last activity
5	4-6	Day (of month) of last Activity
6	Ø,1	
6	2-6	Year of last activity (year-1970)
7	Ø-6	Total Access Count (Read or Write Data)
8	Ø-6	
9	Ø-6	Write Access Count
10	Ø-6	
11	Ø-6	Number of Readings with Correctable Errors
12	Ø-6	
13	Ø-6	Number of Readings with Uncorrectable Errors
14	Ø-6	
15	Ø-6	Serial Number of Video Head Used in Write
16	Ø-6	Alignment Number of Video Head Used in Write

Figure 32 (Continued)

17	Ø-6	Write Retry Count
18	Ø	Number of Write Channel Used in Write
19	1-6	Number of Transport Used in Write

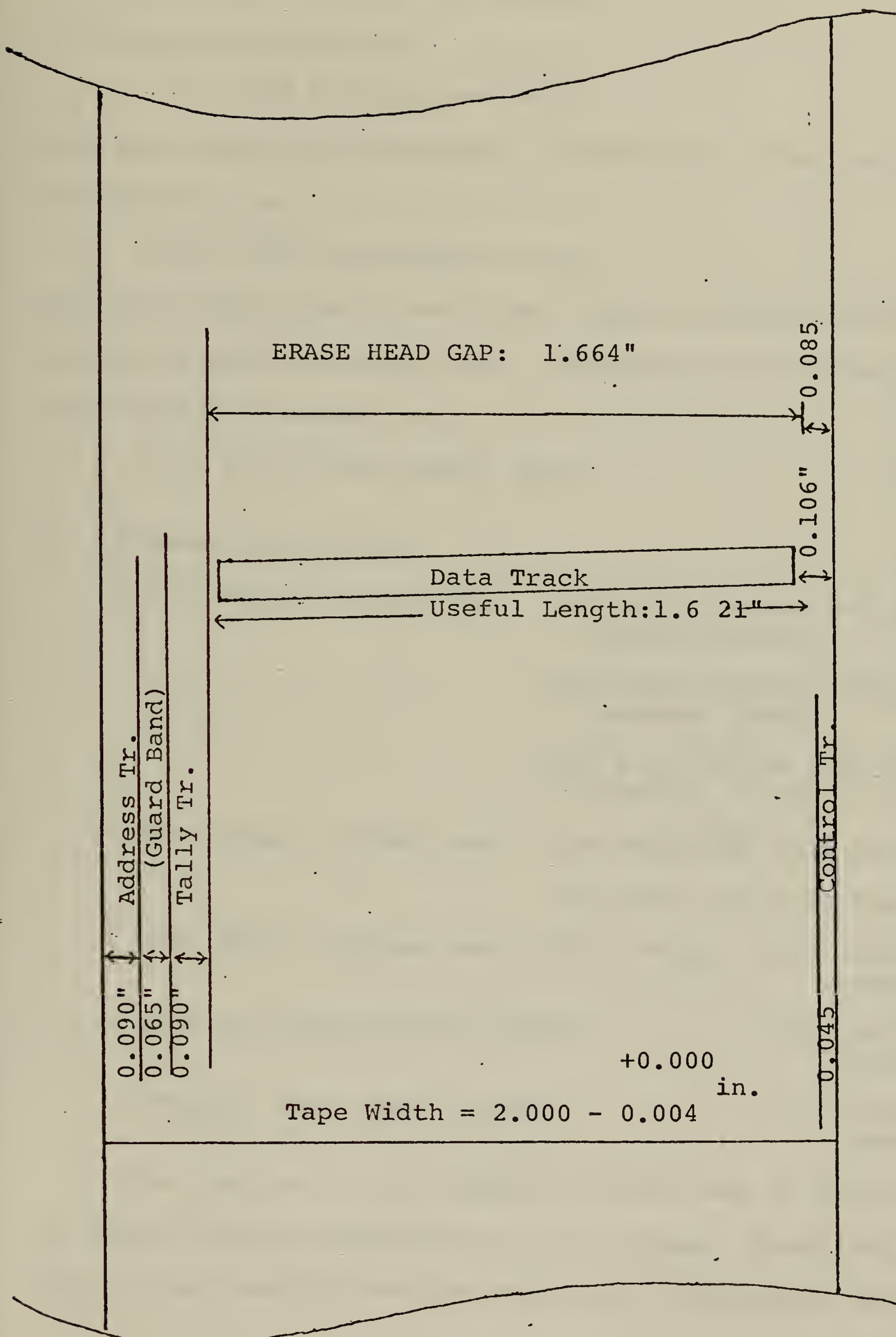


Figure 33 TRACK DIMENSIONS

7800 bits/linear inch of track
189 tracks/inch of TBM tape

This gives bit density:

1.5×10^6 bits/square inch.

All bits recorded redundantly. Therefore, effective data bit density is:

0.75×10^6 bits/square inch.

For every 985 bits of user data, there is another 38 error detection and correction bits. Therefore, the effective user data bit density is:

0.73×10^6 bits/square inch.

B. STORAGE EQUIVALENTS

1 TBM* Memory System Tape	500 Fully Packed 800 bpi Computer Tapes
	250 Fully Packed 1600 bpi Computer Tapes
	2500 Avg. Packed 800/1600 bpi Computer Tapes
1 TBM* Memory System Tape	160 Full 2314 Disc Packs
	55 Full 3330 Disc Packs

1 TBM* Memory System Tape, Fully Packed	\$150	.0000003 cents/bit
1 3330 Disc Pack, Fully Packed	\$750	.00009 cents/bit
1 Computer Tape, Avg. Packed	\$ 10	.00005 cents/bit

Being limited to 2000 reads or writes may at first appear to impose severe limitations on the system. Practically this is not usually the case since most files/data sets are

accessed a few times per day over extended periods of time (very frequently accessed files should be on disc). For example, if a file is read once a day, that particular segment of tape would last approximately 2000 processing days or close to ten years. If the file, on the average, is updated once a day, the tape would last five years. The system keeps track of the number of times each TBM block has been accessed on the Tally track, and raises a block/file move flag when the access count reaches the limit. If the flag is raised, the data set can be moved to another section of the tape. A full TBM tape can be rewritten in two hours in an off-line mode.

Searching over a section of tape does not affect the read/write count. Normal tape quality allows for in excess of 50,000 searches over any segment of tape.

TBM tapes can be replaced for about $150, or if pretested, $250.

C. TRACK FACTS

Data Tracks:

- Each transverse track comprises two (2) segments.
- Each segment (except the first in a block) contains 6 code-words:

5910	(6 x 985)	data bits
228	(6 x 38)	error check bits
192		segment sync bits
6330		Total bits

- Each bit stored in a track occupies: 132 " (0.000132") of track length.

- Scanned at 792 in/s, this recorded wavelength yields a 6 mbit/s data stream.

- Tracks are 3.5 mils (0.0035 in.) in width and are separated by 1.8 mil (0.0018 in.) guard bands. Track spacing is 5.3 mils (0.0053 in.)

- Each data block contains 184 track spaces. 183 of these are used for data storage. The 184th is sacrificed to provide reasonable tolerances for erase start/stop. The loss of one complete track renders one-half of each of two other tracks unusable because of the loss of redundancy. Thus, the usable data space is the equivalent of 182 tracks.

Control Track:

- Each control track cycle occupies the same length of tape as eight data tracks, eight address track bits, or eight tally track bits:

  42.4 mils (0.0424 in.)

- Each block space on tape yields 23 control track pulses and one "Frame Pulse."

- The frequency and period of control track pulses vary with tape speed:

Speed	Frequency	Period
5 in/s	122 Hz	8 ms
83 in/s	1953 Hz	0.5 ms
248 in/s	5859 Hz	170 s
1000 in/s	23.6 kHz	42.4 s

Address Tracks:

- Each block space contains 184 address track bit spaces.

- Two 16-bit words are read from each address block:

  16 bit block address, and 8 bit tape number + 8 bits of the block address

- Each address track bit occupies the same length of tape as one data track:

  5.3 mils (0.0053 in.)

- The rate at which addresses may be read from the tape depends on the tape speed: (approx.)

Speed	Address Interval
5 in/s	190 ms
83 in/s	12 ms
248 in/s	4 ms
1000 in/s	1 ms

Tally Track:

- Each block space contains 184 tally track bit spaces.

- Each block may contain up to 19 bytes of arbitrary information. The remaining four bytes are used for bit and block sync, block checksum, and inter-block gap.

- Each tally information byte includes seven (7) arbitrary information information bits and one parity bit.

- Each tally byte occupies the same tape length as one Control Track cycle or eight Data Track.

- The tally track is written or read at either 5 in/s or 83 in/s. The byte intervals for the two speeds are: (approx.)

  5 in/s - 8 ms
  83 in/s - 0.5 ms

APPENDIX E

TYPICAL FILE UPDATING TECHNIQUES

A. FILE DIRECTORY RETRIEVAL

The TBM* Memory System uses the index sequential method for data retrieval. The host sends a data set request to the TBM SCP (Supervisor Control Processer). The host then puts the user's job on an inactive status and continues with other processing. The SCP takes the data set request and searches its file directory. This search takes only 100 - 400 ms since the file directory is kept on a 3330 disc. The file directory is multi-level with only the first level being a straight sequential search. If the first level is: 1) continuously updated, 2) kept contingously, and 3) not fragmented, then the first level search is fast. The first level gives a location on the disc for the next level of search. After the first level, there is never a need to do a sequential search, since each succeeding level gives another address. When the data set description block is found in the file directory, the tape number, the location on the tape, and the disc space required is kept by the SCP. The disc space required is sent to the host and the search for the data set begins.

The size of the file directory is moderate for a 3330 disc. Even if a large data base of 10^5 files exists with each file address requiring 500 bytes in the file directory, (which should be more than enough), then a total of 5×10^7

bytes would be stored in the file directory, which is less than half of one 3330 disc pack.

B. REDUCE HOST ACTIVITY

Using either single or dual interface buffers, TBM provides exceptionally rapid capability in the file update task. By knowing ahead of time which records are required by the host CPU, TBM can index these records, put the TBM block on disc, and let the host get just the record or records it needs. Updated records which will not fit back into their original space will be placed in overflow locations with their positions noted. When TBM rewrites the disc buffer on tape, the proper sequence of the TBM blocks will be retained. This technique will dramatically reduce channel time, processor time and ultimately job time. For example, if the hit ratio is 1%, then the load on the host I/O channel is reduced by more than a factor of 100.

Illustrative Examples:

Tape In - Tape Out Method: This is the conventional method of updating sequential files by creating a new file in each update process. Logical records are read onto disc where keys are examined. Records with modifications are transferred to the host CPU where processing is performed. Modified records and additions to the file are transferred from the disc through the EDCP in proper sequence and written by TBM. Records to be dropped from the file are simply deleted in the disc.

Fixed length records are handled as illustrated in Figure 34. Records 1, 2, 4, 5, and 7 are read on to disc. Only records to be modified are transferred from disc to the host CPU. When the host is finished processing, the modified records and the new records are written to disc. The inactive records never leave the disc. The records are written in sequence back to the TBM* Memory System.

Variable length records that may change size are reformatted on disc and are written out in sequence as illustrated in Figure 35. In this example, the updated records, #2 and #5, are expanded by the host CPU. The added portions of the updated records and the new records, #3 and #6, are written to disc by the host CPU. The records are written in sequence on the TBM* Memory System.

The same example as above is illustrated in Figure 36 for the dual buffer case.

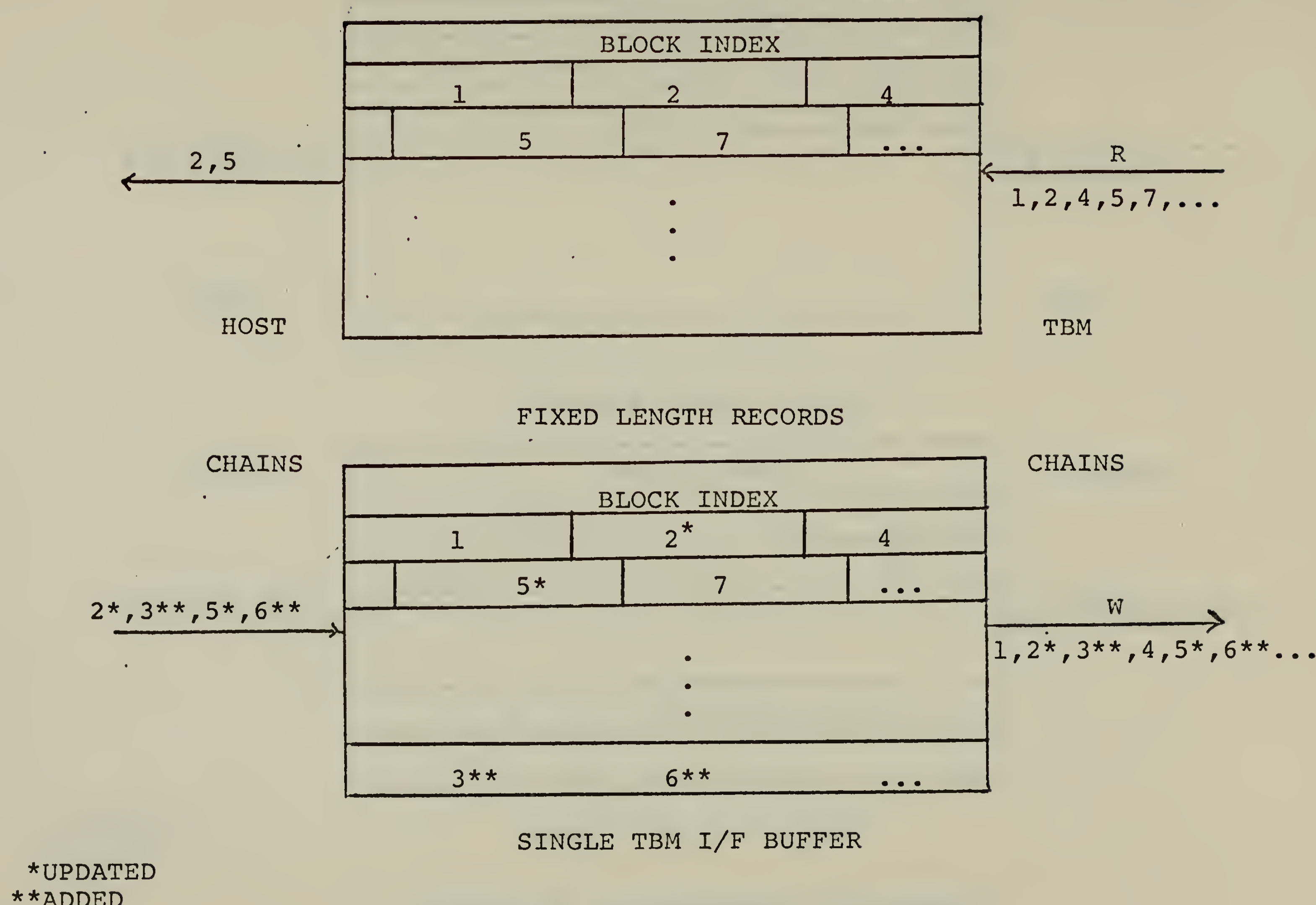


Figure 34 ADD/UPDATE - IN PLACE

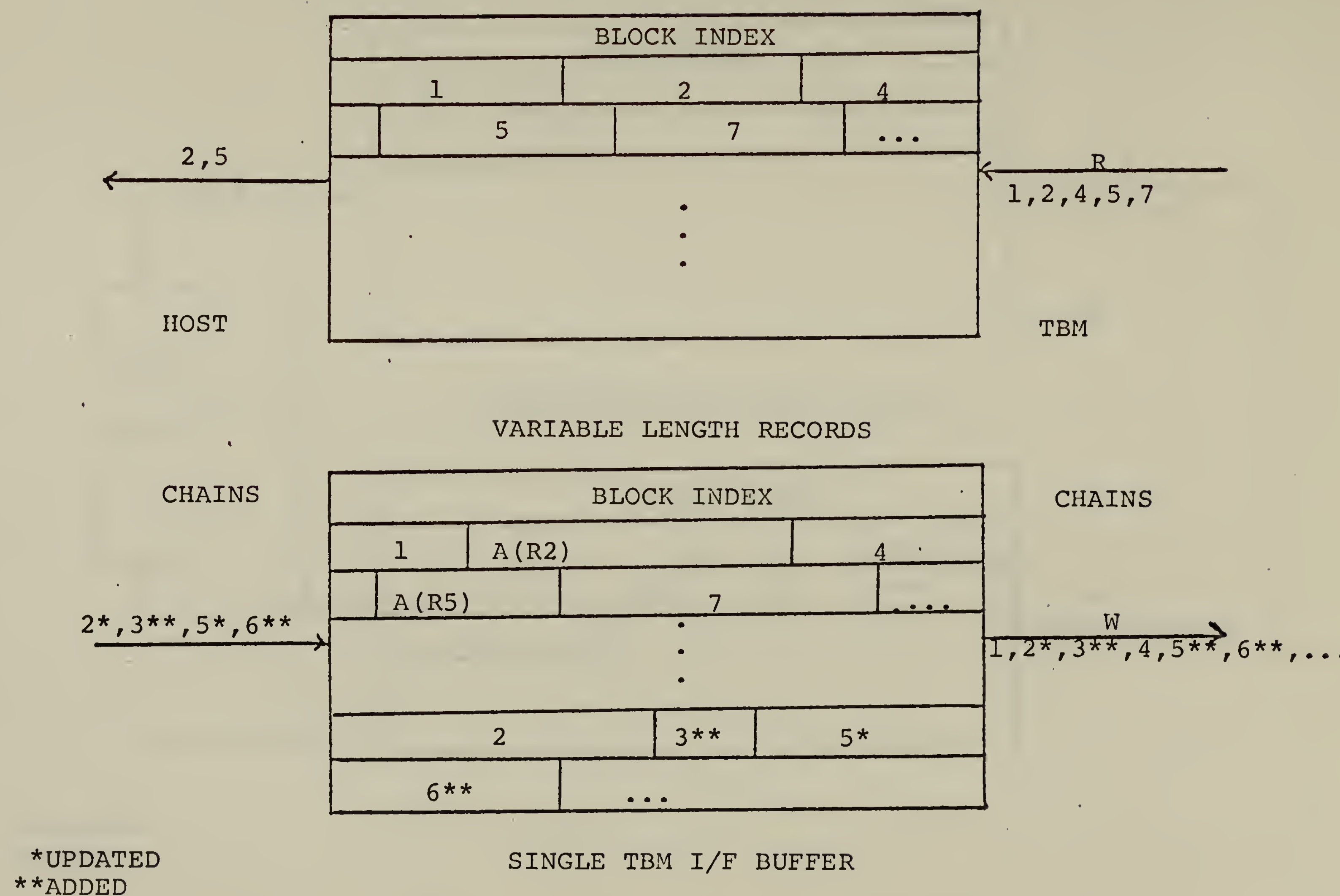
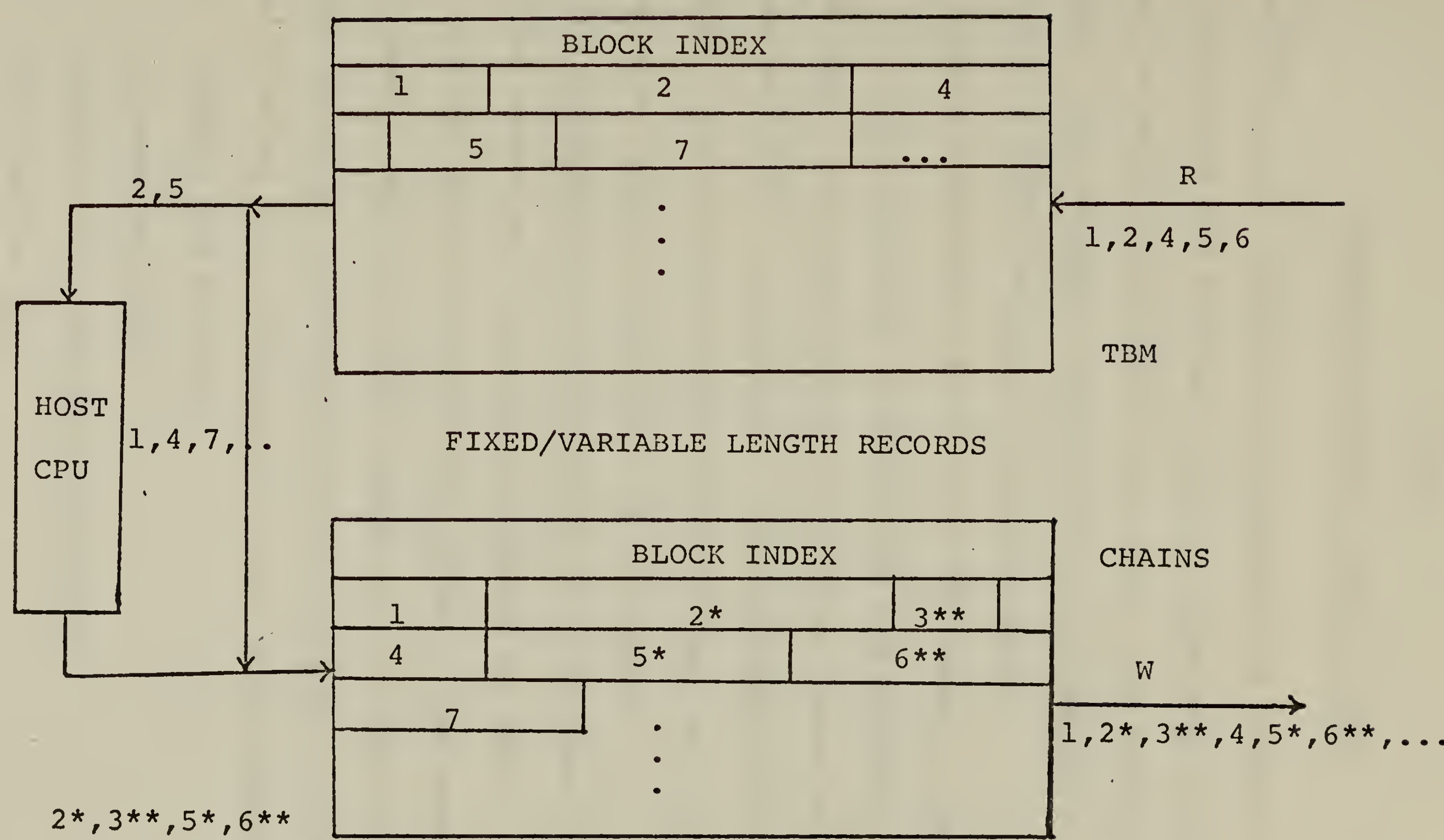


Figure 35 ADD/UPDATE - IN PLACE

ADD/UPDATE - DUAL BUFFERS



*UPDATED
**ADDED

Figure 36 ADD/UPDATE - DUAL BUFFERS

APPENDIX F

TBM APPLICATIONS FOR CP USERS

Normally the quicker the response the better the system is. The trade off between response time and file storage system is usually dependent upon cost. In certain applications, like terminal response in an interaction system, i.e., CP, a quick response time is a must.

Certain configurations of the TBM* Memory System could meet the quick response criteria and yet maintain the files on TBM tape. The statistics of terminal usage may help to explain why this is possible. The following data does not represent any one particular installation, but is an approximation to a typical population of CP users.

In a representative population of 1000 users, the number of active users each day varies between 50 and 75. These users pay for their CPU time and normally the amount of core memory used. Naturally they would want to sign on, run their program, and sign off as quickly as possible. After the initial running of a program, the user will normally:

1) Correct his program;
2) Prepare new data;
3) Prepare for a different program;
4) Quit for the day.

Conclusions about CP users are as follows:

1) Although the user population is large, the number of daily users is small.

2) If a user signs on once during the day, he will probably sign on again.

3) A typical user's CPU time per sign-on is from 1/2 minute to 5 minutes.

4) The average file stored in the TBM* system is approximately 512000 bytes with a range from 128000 bytes to 2.5 megabytes.

5) The average number of data sets per job is two.

6) The activity of daily users is heaviest during working hours as expected, (see Figure 37).

7) The average number of megabytes required to store all of the personal files of each user is 1.2. See Figure 38 for the distribution.

8) The maximum rate of initial sign-on is 20 per hour.

The main stipulation is to meet the quick response requirement. The major time factors in the TBM* system are:

1) the wait to get a transport driver;

2) the search and read time.

A proposed method, that was tested by the simulation model, is to allocate one or two transport drivers for CP files only. The entire population of interactive files would be stored on TBM tape but spread out over as many tapes as necessary so as to allow CP files to be stored only on one-seventh of each tape. This has the effect of reducing search times to 3.5 seconds. See Figure 39 for a comparison of normal search and CP search.

The initial sign-on by the user would have TBM* transfer all or a percentage of the user's files to disc and left on the disc for the remainder of the day. For initial sign-on, it might take up to one minute to load the user files. After initial sign-on, the files would be stored on disc and normal CP interface would take place. This saves disc space by not keeping all the inactive user's files on disc, yet allows relatively quick access by active CP users.

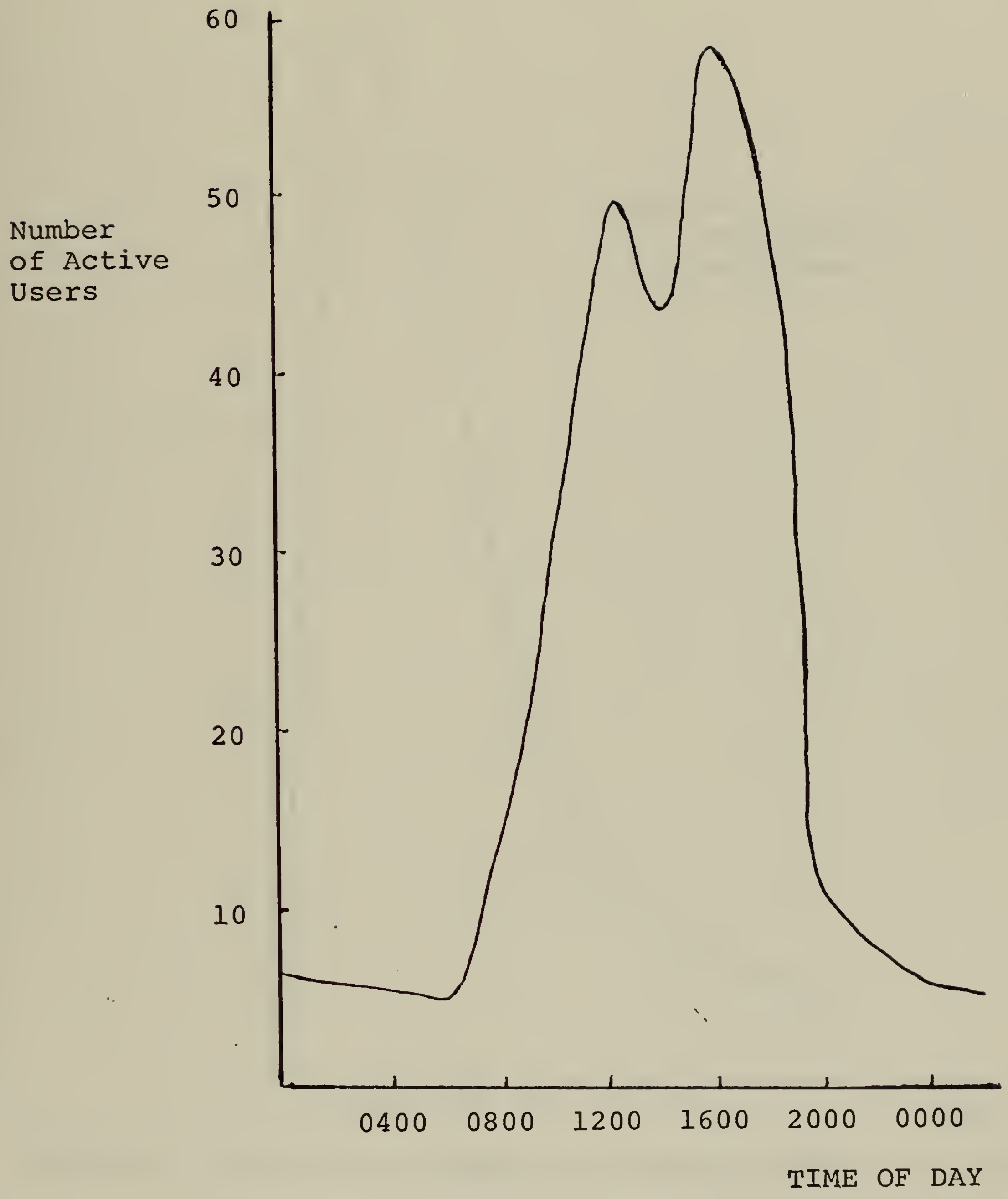
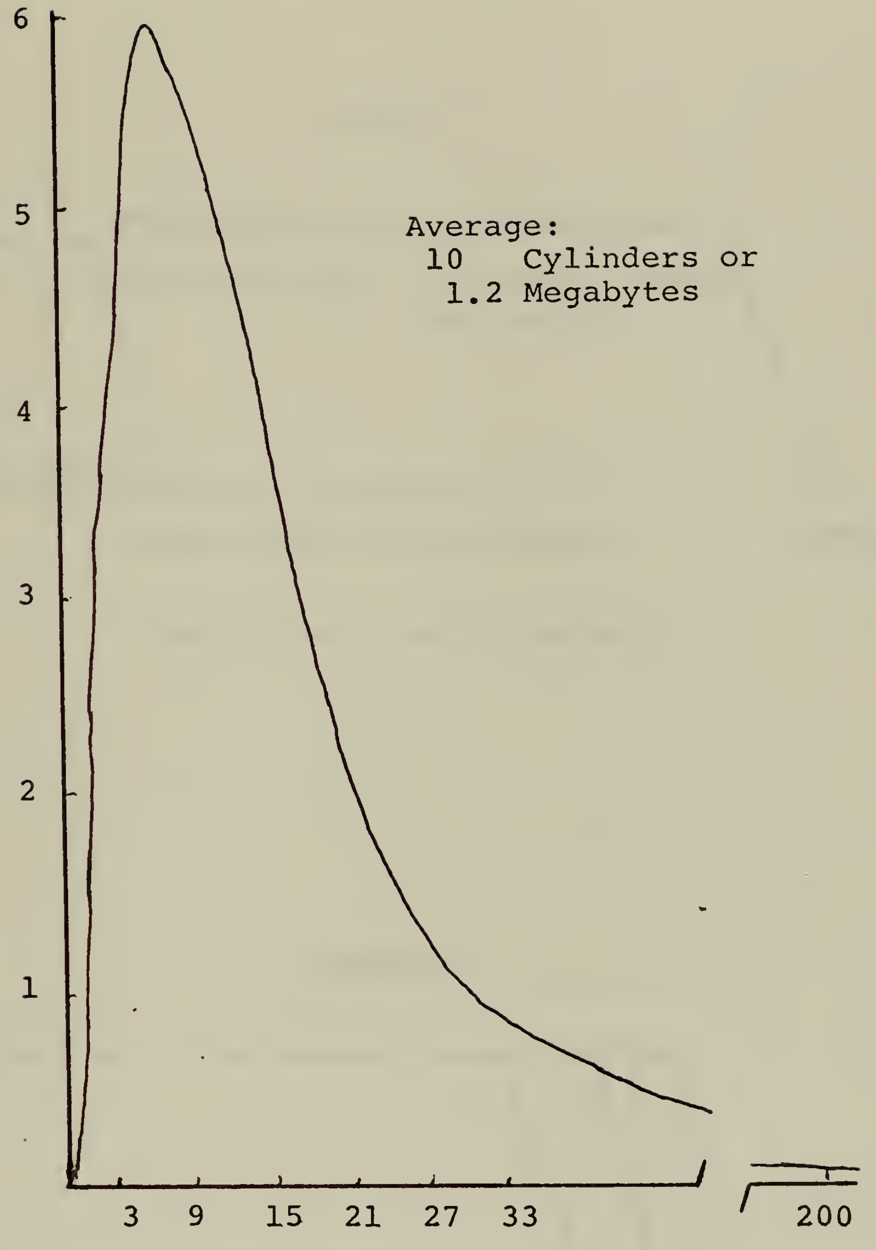
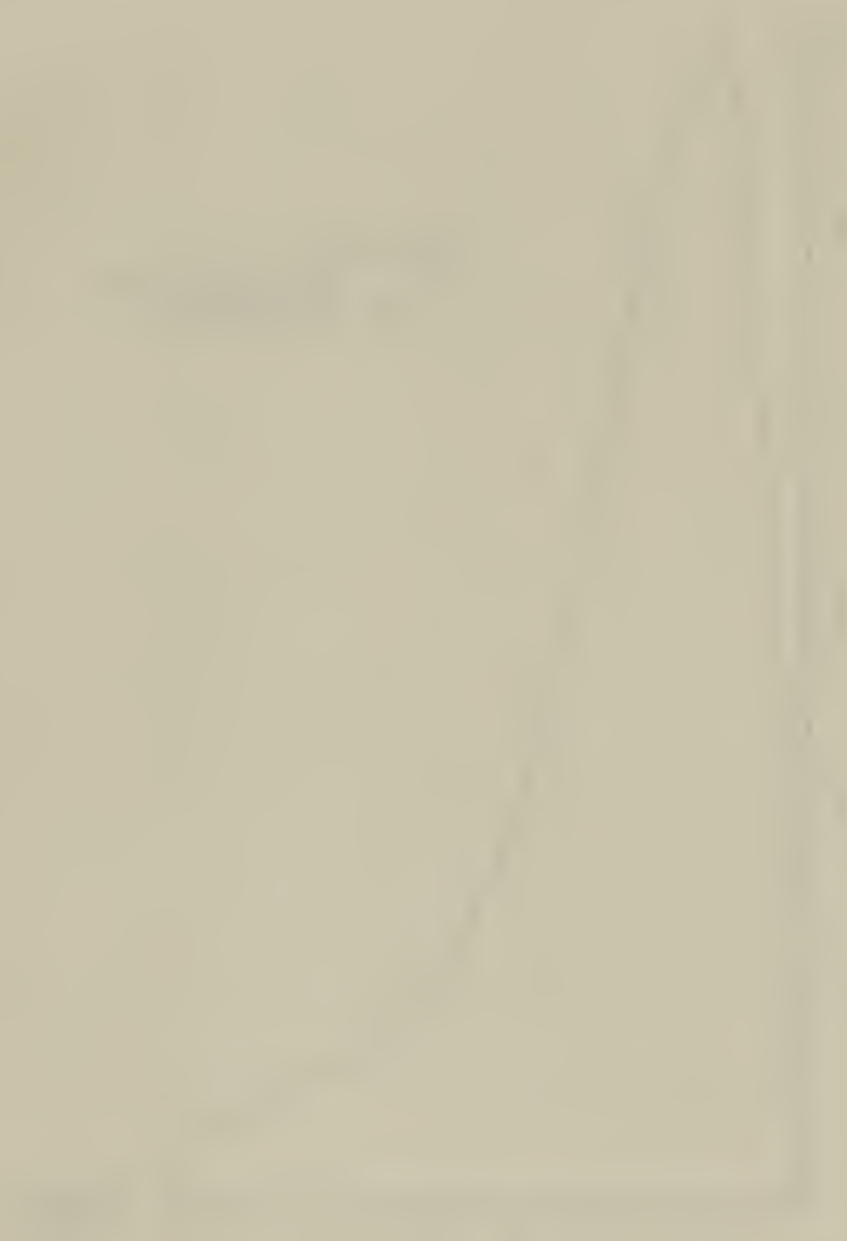


Figure 37 DAILY TERMINAL ACTIVITY

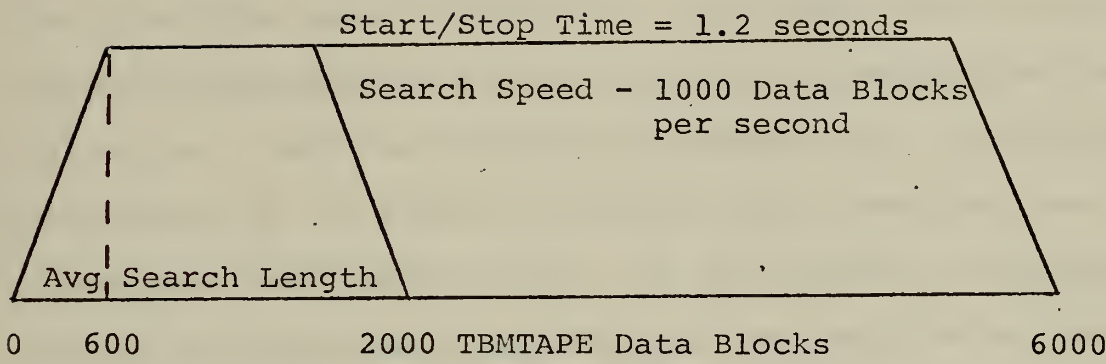


NUMBER OF 2314 DISC CYLINDERS OF PERSONAL STORAGE PER CP USER

Figure 38 DISTRIBUTION OF PERSONAL CP USERS FILES

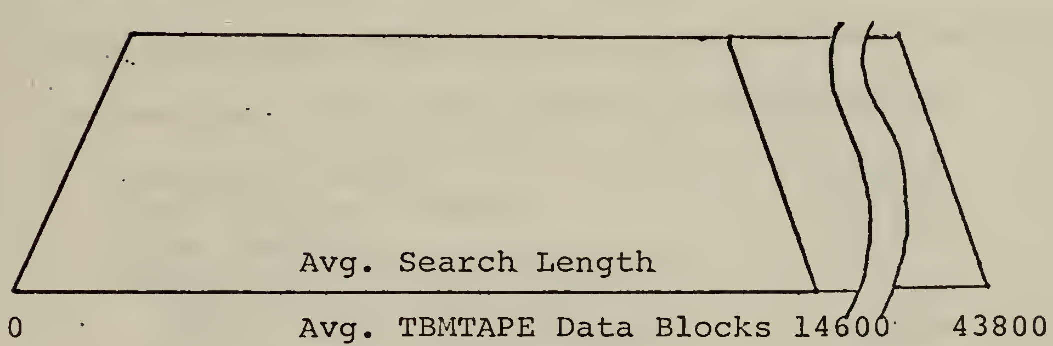


CP/CMS



T(Search Avg.) = 3.5 seconds

STANDARD



T(Search Avg.) = 16.5 seconds

Figure 39 CP AND BATCH TAPE LAYOUT

APPENDIX G

COSTS

A. EQUIPMENT COSTS

This section contains rough cost estimates that were obtained from Ampex Corporation. The TBM* Memory System is normally examined for its suitability to the application and situation. Very few mass storage systems are available on the market at the present time and each system has its advantages and disadvantages. If two systems are adaptable to the mass storage requirements of an installation, then competitive bidding usually is the method of winning the contract. Thus there is no open market price for the TBM system, only rough estimates based on very limited experience. The minimum TBM* Memory System has an initial installed cost of approximately $500,000. This is the minimum system which will hold approximately 10^{11} bits of user data on-line. The modularity of the TBM system allows the installation to build upon this system in any manner to best suit their needs. The cost per additional device is estimated as:

Transport Driver	$75,000
Dual Transport Module	75,000
Data Channel	75,000
Supervisory Control Processor	
(PDP-11 with Equipment)	75,000
(PDP-11 without Equipment)	30,000
External Data Channel	
Processer	55,000
TBM tape	
Not pre-tested	150
Pre-tested	250

Ampex does not sell the 3330 discs or the channel simulator which are assumed to be in the installation's environment.

The approximate cost, depending on maximum on-line storage, is given as follows:

On-Line Storage (Bits)	Equipment	Cost (Installed)
10^{11}	TD, DTM, DC, SCP, EDCP	$ 500,000
10^{12}	TD, 11 DTM, DC, SCP, EDCP	1,250,000
3×10^{12}	6 TD, 36 DTM, 3 DC, 2 SCP, 2 EDCP	3,250,000

See the maximum system in Figure 2.

B. MAINTENANCE COSTS

The maintenance of the TBM* Memory System is normally done under contract with Ampex, although Ampex will train the host's maintenance people. An estimate of the maintenance cost is 1% of the initial installation costs per month.

C. MEDIA COSTS

The cost of a TBM tape is less than $150.00. The capacity of a tape is in excess of 46 billion bits resulting in a cost per bit of approximately 3×10^{-7} cents. Reusing the media a thousand times reduces the cost per bit by the same factor when comparing systems using non-erasable media, (see Figure 40).

The media cost (for tapes) for storing a trillion bits on TBM tape is less than 4000 dollars as compared to more than 100,000 dollars using fully packed conventional computer tapes.

Figure 40

STORAGE DEVICE COMPARISONS

	Unit Price ($)	Unit Capacity (M Bits)	Cost per Bit (Cents/Bit)
TBMTAPE Transport	75K	4.6×10^{10}	16.8×10^{-5}
TBM 330 Disc Drive	15K	8×10^{8}	1.9×10^{-3}
TBM 3420/7 Tape Drive	20K	3×10^{8}	6.7×10^{-3}
TBMTAPE Media	150	4.6×10^{10}	2.18×10^{-7}
TBMTAPE Media	250 (1)	4.6×10^{10}	5.3×10^{-7}
TBM 3336 Disc Pack	750	8×10^{8}	9.4×10^{-5}
1600 bpi Tape	10	2×10^{7}	5×10^{-5}

(1) Pre-tested

APPENDIX H

FIRST CUSTOMER RESULTS

The first customer to install a version of the TBM* Memory System is connected with the D. O. D., but the name and location are classified. The system bought was a special purpose system. The hardware purchased included:

1. 36 dual tape modules

2. 2 data channels

3. 4 transport drivers

4. DEC PDP-9 Computer (Supervisory Control Processer):

   16K x 18 bit Core Memory
   Extended Arithmetic Element (EAE)
   Automatic Priority Interrupt (API)
   Real Time Clock
   300 cps Paper Tape Reader
   50 cps Paper Tape Punch
   KSR 35 Console Teleprinter
   Multi-Station Teletype Control
     w/one KSR 33 Teleprinter
   Four TU55 DECtape Transports, w/controller

5. Extended Core Memory (ECM):

   32K x 32 bit External Core Memory
   Interface to PDP-9, program I/0, data channel I/0
   Interface to TBM* Memory System Data Channels
     (4 part)

6. Special Interface to TCP's of TBM* Memory System

7. I/0 Bus connects to DA10 (PDP-9 to PDP-10)
   interface to SCP

Note that the SCP was not a PDP-11 and that ECM was used in place of an EDCP. See Figure 41.

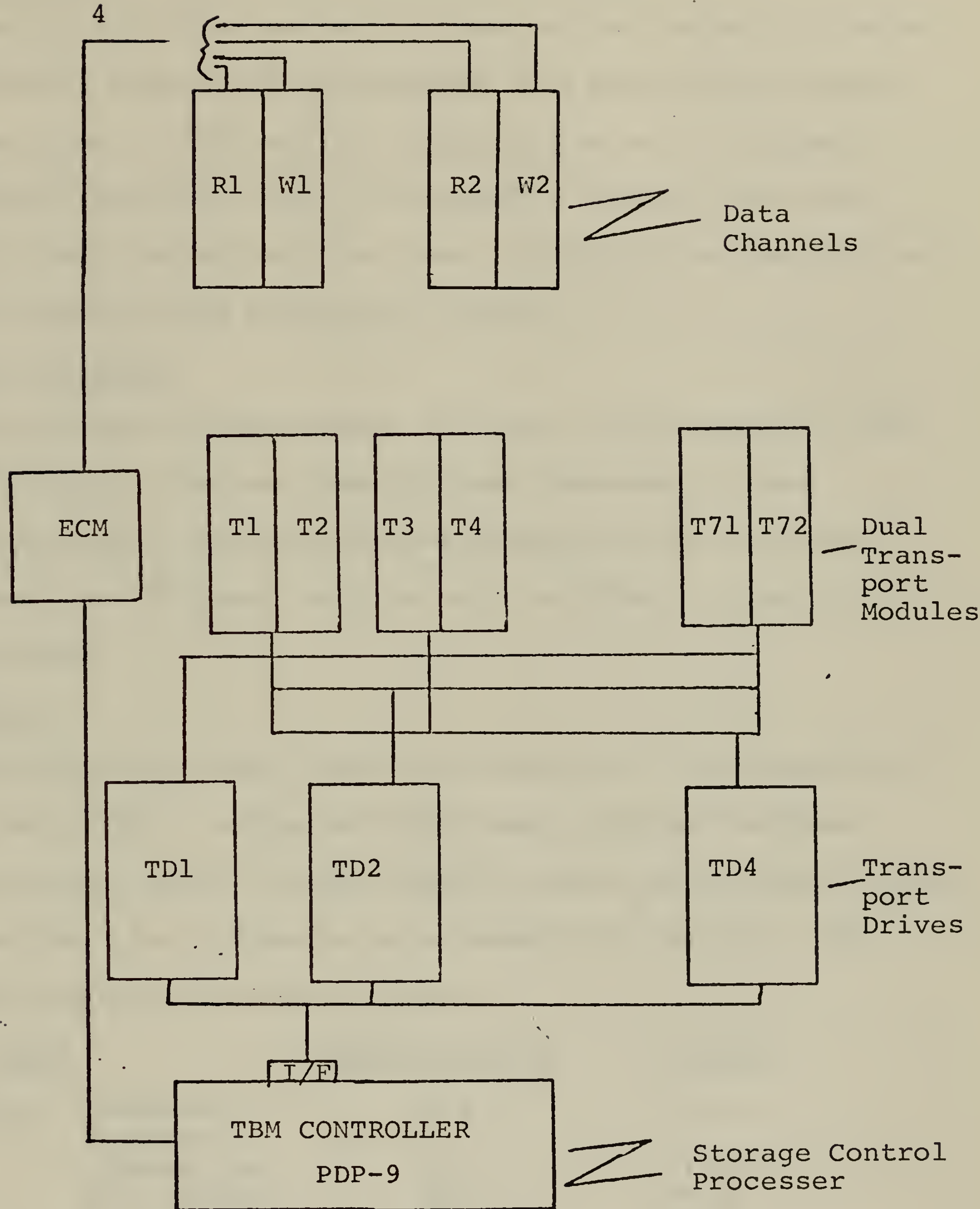


Figure 41 TBM* MEMORY SYSTEM AS CONFIGURED FOR THE FIRST CUSTOMER

Maintenance

The installation wanted its own people trained in system maintenance instead of contracting with Ampex to have maintenance done. This required training twelve to fifteen qualified people in order to maintain a twenty-four hour surveillance, seven days per week. The training period took approximately three months full time.

Initial Testing

The System was pre-tested at Ampex and re-tested at the installation. The test results were required to meet specifications. In 220 hours of testing within an eleven day span the following results were recorded as shown in the summary.

Summary

During these tests, the total number of bits transferred was 1.68×10^{12}. All major TBM* Memory System functions, such as read, write, erase, search, update were tested during the period. The following is a summary of results. The system proved to be very reliable.

ITEM		SPECIFICATION	RESULTS
Up Time:	Transport	90%	98.5%
	Transport Controller	90%	97.0%
	Data Channel	90%	99.4%
	System	90%	98.3%
Rereads per 1000		2.5	0.37
Demarked Blocks		1%	0.027
Data Transfers per Block		1200	3200 $\pm$ 850
Unrecoverable Errors		1 in 2×10^{10}	0 in $\bar{1}.4 \times 10^{12}$

Level A Acceptance Test Summary
Period: Day 1 - Day 11 (220 hours of testing)

Blocks Written:	Multi-block Mode	100,620
	1-block mode	92,524
	Total	193,144
Blocks Read:	Multi-block Mode	1,062,390
	1-block mode	363,739
	Total	1,426,129
Re-Read Rate (#/1000)	-Multierror Detected:	
	Multi-block Mode (231)	0.22
	1-block Mode (302)	0.83
	Overall (533)	0.3711000 (weighted average)
Unrecoverables - cannot get information		0
Unrecoverable Procedures - (could not get by re-read. Had special process, i.e., clean tape		2
Undetected Errors		0
# Selective Updates		12,355
% Bad Erase/Writes (27/95234)		0.028%
T5 & T13 Point-Point searches		15,513
Command Failure Rate:	Multi-block Mode	(2399/31694 = 7.57%
	1-block Mode	(7782/621775)=1.25%
	Overall (Weighted)	(10181/653469)=1.56%
Demarked Blocks on Tape:		
	Pre-test	(27/98170)=0.028%
	Post-test	(22/1109907)=0.002%

APPENDIX I

```
                SIMULATION OF THE TBM* MEMORY SYSTEM

*          PROGRAMMED BY ROBERT B. FRAZIER FOR HIS THESIS
*          AS A CANDIDATE FOR A MASTERS OF SCIENCE IN
*          OPERATIONS RESEARCH / SYSTEMS ANALYSIS

*              THE PROGRAMMING LANGUAGE IS GENERAL PURPOSE SIMULATION
*          SYSTEM (GPSS).  NO DATA IS ALLOWED THEREFORE ALL VARIABLES
*          ARE CONTROLLED BY CHANGING THE PROGRAM.  THIS PROGRAM INCLUDES
*          MANY OF THE POSSIBLE CONDITIONS OF THE SYSTEM BUT SOME
*          CONDITIONS ARE MUTUALLY EXCLUSIVE (IE. DATA SET ARRANGEMENT
*          BY ACTIVITY VERSE SIZE).  THEREFORE THIS PRINTOUT DOES NOT
*          BEGIN TO SHOW ALL THAT WAS TESTED.

*          THE TIME UNIT IS TENTHS OF A SECOND

*          GENERATE JOBS WITH THE FOLLOWING VARIABLES

*              1  INTERARRIVAL MEAN
*              2  INTERARRIVAL DISTRIBUTION
*              3  PRIORITY
*              4  NUMBER OF DATA SETS PER JOB
*              5  SIZE OF EACH DATA SET
*              6  LOCATION OF DATA SET

*          PARA 1 = LOCATION OF TAPE HEAD
*          PARA 2 = DATA SET SIZE IN TBMBLOCKS
*          PARA 3 = JOB PRIORITY
*          PARA 4 = DATA SETS REQUESTED PER JOB
*          PARA 5 = TBMTAPE THAT DATA SET IS ON
*          PARA 6 = LOCATION OF DATA SET ON TBMTAPE
*          PARA 7 = NO. OF TAPE DRIVES AVALIABLE
*          PARA 8 = TAPE HEAD LOCATION

*          SAVEVALUES  1 THRU 36 = STORE READER HEAD LOCATION
*          SAVEVALUES 37 THAN 72 = # OF REQUESTS FOR THIS SEARCH
*          SAVEVALUE 74 = # OF TRANSPORT DRIVES IN THE SYSTEM
*          SAVEVALUE 75 = A COUNTER
*          SAVEVALUE 76 = MINIMUN NUMBER OF JOBS BEFORE SCP RELEASES FOR
*                         PROCESSING
```

```
*          QUEUES 1 THRU 36 ARE TIME OF SEARCH PLUS READ
*          QUEUE # 37 = TIME A REQUEST IS IN THE TBM SYSTEM
*          QUEUE # 38 = TIME A TOPPLATE WAITS FOR A TRANSPORT DRIVE
*          QUEUE # 39 = WAIT FOR DISC CONTROLLER
*          QUEUE # 40 = WAIT FOR SUPERVISORY CONTROL PROCESSOR
*          QUEUE # 41 = WAIT FOR DATA CHANNEL

*          STORAGE 1 IS FOR DATA CHANNEL
*          STORAGE 2 IS FOR DISC CONTROLLER
*          STORAGE 3 IS FOR SUPERVISORY CONTROL PROCESSOR

 1      FUNCTION   RN6,D5       FUNCTION FOR PRIORITY DISTRIBUTION

0.2,1/0.4,2/0.6,3/0.8,4/1.0,5

 2      FUNCTION   RN6,C13      FUNCTION FOR DATA SET SIZE IN TBM BLOCKS

0.0,1/0.21,7/0.44,15/0.65,23/0.76,31/0.84,38/0.90,46/0.94,54
0.97,61/0.99,68/0.996,77/0.998,114/1.0,231

 3      FUNCTION   RN6,C24      JOB INTERARRIVAL DISTRIBUTION

0.0,0.0/0.1,0.104/0.2,0.222/0.3,0.355/0.4,0.509/0.5,0.69
0.6,0.915/0.7,1.2/0.75,1.38/0.8,1.6/0.84,1.83/0.88,2.12
0.9,2.3/0.92,2.52/0.94,2.81/0.95,2.99/0.96,3.2/0.97,3.5
0.98,3.9/0.99,4.6/0.995,5.3/0.998,6.2/0.999,7/1.0,8

 4      FUNCTION   RN5,D3       DISTR - NO. DATA SETS PER JOB

0.333,2/0.667,5/1.0,8

 5      FUNCTION   RN6,D5       LOCATION PICK AN ONLINE TBMTAPE

0.2,1/0.4,2/0.6,3/0.8,4/1.0,5

 6      FUNCTION   RN6,C45      FUNCTION TO ARRANGE DATA SETS BY ACTIVITY

0.0,0/0.013,1000/0.018,2000/0.02,3000/0.03,4000/0.04,5000/0.05,6000
0.06,7000/0.07,8000/0.08,9000/0.10,10000/0.12,11000/0.14,12000
0.16,13000/0.18,14000/0.21,15000/0.24,16000/0.27,17000/0.31,18000
0.34,19000/0.38,20000/0.42,21000/0.46,22000/0.50,23000/0.54,24000
0.58,25000/0.62,26000/0.66,27000/0.69,28000/0.72,29000/0.75,30000
0.78,31000/0.81,32000/0.84,33000/0.86,34000/0.88,35000/0.90,36000
0.92,37000/0.93,38000/0.95,39000/0.96,40000/0.97,41000
0.98,42000/0.99,43000/1.00,44000
```

```
 9     FUNCTION    P1,C13       SEARCH TIME FOR LESS THAN 1230 BLOCKS

0,0.0/100,0.629/200,0.97/300,1.2/400,1.38/500,1.55/600,1.7
700,1.83/800,1.96/900,2.08/1000,214/1100,2.3/1200,2.4

*
*          PROGRAMING TECHNIQUE TO INITIATE LINK BLOCKS
*
       GENERATE    0,,,1          PROGRAMMING TECHNIQUE FOR PRIORITY
       TRANSFER    ,ALINK         GO TO PRIORITY LINK
       GENERATE    0,,,6          PROGRAMMING TECHNIQUE FOR TRANSPORT DRIVE
       ASSIGN      5,V10          GET VALUE OF ALL TAPES
       SAVEVALUE   75,P5          USE FOR INCREMENTING
       TRANSFER    ,ELINK         TO GO THRU LINK FOR TRANSPORT DRIVES
*
*          GENERATE JOB REQUESTS - ASSIGN JOB PARAMETERS
*
       GENERATE    600,FN3,,,,,F       GENERATE 1 JOB REQUEST PER MINUTE
       ASSIGN      3,FN1          P3 = PRIORITY OF JOB
       ASSIGN      4,FN4          P4 = # DATA SETS PER JOB
       TABULATE    3              TO ENTER DATA SETS PER JOB STATISTICS
       SPLIT       P4,PROUT       CREATE P4 DATA SET REQUESTS FOR THIS JOB
 PROUT ASSIGN      2,FN2          P2 = DATA SET SIZE
       ASSIGN      5,FN5          P5 = WHICH TOPPLATE THE DATA SET IS ON
       ASSIGN      6,FN6          P6 = LOCATION ON TBMTAPE
       TABULATE    1              TO ENTER PRIORITY STATISTICS
       TABULATE    2              TO ENTER DATA SET SIZE STATISTICS
       TABULATE    4              TO ENTER TBM TAPE USAGE STATISTICS
       TABULATE    5              TO GATHER DATA SET LOCATION ON TAPE STATS
*
*          ENTER THE SYSTEM - QUEUE ON PRIORITY - LOOK UP ADDRESS
*          IN THE FILE DIRECTORY ON DISC
*
       QUEUE       37             ENTER THE TBM SYSTEM
 ALINK LINK        37,P3,PRNEX         QUEUE ON PRIORITY
 PRNEX TEST GE     CH37,X76            WAIT UNTIL ENOUGH JOBS RECEIVED
       UNLINK      37,CONTA,ALL        RELEASE ALL JOB REQUESTS FOR PROCESS
       TRANSFER    ,PRNEX         GO TO PRNEX AND WAIT
 CONTA QUEUE       39             WAIT FOR DISC CONTROLLER
       ENTER       2              GET A DISC CONTROLLER
       DEPART      39             LEAVE QUEUE FOR DISC CONTROLLER
       QUEUE       40             QUEUE FOR SCP FILE DIRECTORY SEARCH
       ENTER       3              GET A SUPERVISOR CONTROL PROCESSOR
       DEPART      40             LEAVE QUEUE TO BE SEARCHED
       ADVANCE     1              TIME FOR SCP TO FIND ADDRESS IN FILE
       LEAVE       2              FREE A DISC CONTROLLER
       LEAVE       3              FREE A SUPERVISORY CONTROL PROCESSOR
```

```
*          ENTER QUEUE FOR TOPPLATE - QUEUE BY PHYSICAL LOCATION
*          OF DATA SET ON TAPE.  WAIT UNTIL ENOUGH REQUESTS EXIST
*          THEN GET A TRANSPORT DRIVER
*
 ELINK LINK        P5,P6,NEXTA ARRANGE THE DS REQUESTS SEQUENTIALLY
 NEXTA TEST GE     CH*5,1       IS THERE X REQUESTS FOR A TOPPLATE -
                                YES, INITIATE SEARCH;  NO, WAIT
       TEST L      P5,X77,CONTG     IF WRITE TAPE GO AROUND WRITE TEST
*
*          CHECK TO SEE IF ENOUGH DISC SPACE IS AVAILABLE - IF NOT
*          THEN INITIATE WRITE
*
       TEST LE     S4,X73,DWRIT     IF DISC NEAR FILLED - INITIATE WRITE
 CONTG GATE LR     P5           IS TAPE BUSY, YES - PROCESS OTHER TAPES
       LOGIC S     P5           FLAG TOPPLATE AS BUSY
       QUEUE       38           QUEUE FOR TRANSPORT DRIVE
       SEIZE       38           ONLY ONE REQUEST FOR A TD AT A TIME
       ASSIGN      7,V8         DECREMENT COUNTER FOR TRANSPORT DRIVE
       SAVEVALUE   74,P7        STORE THE NEW VALUE
       TEST GE     X74,0        IS A TRANSPORT DRIVE AVALIABLE - NO, WAIT
       RELEASE     38           ALLOW ANOTHER REQUEST FOR A TAPE DRIVE
       DEPART      38           LEAVE QUEUE FOR TRANSPORT DRIVE
       ADVANCE     9            TIME FOR SWITCHING CONNECTION
       ASSIGN      7,CH*5       NO. OF REQUESTS TO BE SEARCHED FOR
       INDEX       5,6          COMPUTE WHERE TO STORE
       SAVEVALUE   *1,P7        STORE # OF REQUESTS
       UNLINK      P5,CONTB,ALL     COMBINE ALL REQUESTS FOR A TOPPLATE
       TRANSFER    ,NEXTA       GO AND WAIT FOR NEXT BATCH OF REQUESTS
 CONTB TEST L      P5,X77,BACK      WON'T HAVE TO SEARCH WRITE TAPE
*
*          HAVE A TRANSPORT DRIVER.  BEGIN SEARCH TO FIRST LOCATION
*          SEARCH AND READ ALL QUEUED REQUESTS ONE AT A TIME
*          FIND LOCATION OF FIRST REQUEST - POSITION FOR READ OR WRITE -
*
       ASSIGN      1,X*5        GET THE PRESENT HEAD LOCATION
       ASSIGN      1,V4         COMPUTE THE SEARCH DISTANCE IN BLOCKS
       TEST GE     P1,0,POS     MAKE SEARCH DISTANCE POSITIVE
 BACK  ASSIGN      8,V7         PARA 8 = HEAD LOCATION AFTER READ
       SAVEVALUE   *5,P8        STORE HEAD LOCATION FOR NEXT TRANSACTION
       QUEUE       P5           TIME TOPPLATE IS BUSY - SEARCH AND READ
       SEIZE       P5           ONE LOCATION FOUND AND READ
       TEST L      P5,X77,CONTD  DOES NOT NEED TO SEARCH THE WRITE TAPE
       TEST LE     P1,K1230,CONTC  IF SEARCH DIST DOESN'T ALLOW 1000 IPS
       ADVANCE     10,FN9        THEN TIME TO SEARCH IS SLOWER
       TRANSFER    ,CONTD
 CONTC ADVANCE     V2           SEARCH TIME WHEN 1000 IPS IS REACHED
```

```
*          GET A CATA CHANNEL, EDCP, ANC A DISC CCNTRCLLER - THEN READ
*
 CONTD CUEUE       41              QUEUE FOR DATA CHANNEL
       ENTER       1               GET A DATA CHANNEL
       DEPART      41              DEPART QUEUE FOR DATA CHANNEL
       CUEUE       39              WAIT FOR DISC CONTRCLLER
       ENTER       2               GET A DISC CONTRCLLER
       DEPART      39              LEAVE QUEUE FOR DISC CCNTRCLLER
       TEST L      P5,X77,WRITE        WRITE TAKES LCNGER THAN READ
       ENTER       4,P2            GET ASSIGNED DISC SPACE
 RREAD ACVANCE     V3,             READ TIME - FROM TAPE THRU DATA CHANNEL
                                   THROUGH ECCP AND CN TC DISC
       TRANSFER    .001,WBACK,RREAD  PERCENT OF REREADS DUE TO ERROR
 WBACK LEAVE       2               FREE A DISC CONTRCLLER
       LEAVE       1               FREE A CATA CHANNEL
       RELEASE     P5              FINISHED WITH THAT LOCATION
       CEPART      P5              ONE LESS REQUEST TO BE PROCESSED
*
*          FINISHED WITH TRANSPORT CRIVE? - YES, RELEASE IT -
*          NO, BEGIN SEARCH FCR NEXT REQUEST OF BATCH
*
       INDEX       5,6             COMPUTE WHERE TO FIND STORED VALUE
       ASSIGN      7,V11           COUNTER:  ALL QUEUEC REQUESTS COMPLETED
       SAVEVALUE   *1,P7           STORE THE NEW VALUE
       TEST E      X*1,0,AROUN     IF FINISHED RELEASE THE TRANSPORT DRIVE
       ASSIGN      7,V9            INCREMENT CCUNTER FCR TRANSPORT CRIVES
       SAVEVALUE   74,P7           STORE THE NEW VALUE
       LCGIC R     P5              TOPPLATE NOT IN USE
 AROUN TEST L      P5,X77,EXIT          ENC CF WRITE TRANSACTION
       DEPART      37              TIME REQUEST IS IN THE TBM SYSTEM
*
*          THE HOST PROCESSING TIME
*          X PERCENT OF HOST REQUESTS ARE READ ONLY
*          1-X PERCENT OF INITIAL REQUESTS RESULT IN UPDATES TO THE DS
*          QUEUE UPDATES ON DISC UNTIL WRITE INITIATED
*
       ACVANCE     V1              HOST PROCESSING TIME
       TRANSFER      .750,CONTE,EXIT   PERCENTAGE CF READ CNLY REQUESTS
 CONTE GATE LS     37              QUEUE UPDATES ON DISC UNTIL WRITE SIGNAL
       LCGIC R     P5              FLAG TOPPLATE AS NCT BUSY
       ASSIGN      5,X77           P5 = WRITE TAPE #
       TRANSFER    ,ELINK          GO TO QUEUE FOR TRANSPORT DRIVES
 EXIT  LEAVE       4,P2            FREE DISC SPACE
       TERMINATE   1               HOST FINISHED WITH CATA SET
 WRITE ACVANCE     V6              WRITE TIME
       TRANSFER    ,WBACK          RETURN
 PCS   ASSIGN      1,V5            MAKE SEARCH DISTANCE POSITIVE
       TRANSFER    ,BACK           RETURN FCR PROCESSING
```

```
DWRIT LCGIC S     37              SET SWITCH TO ALLCW WRITE
      TRANSFER    ,CONTG          RETURN FCR PROCESSING

      INITIAL     X1,20000        SET INITIAL HEAD LCCATION TO MIDPT
      INITIAL     X2,20000        SET INITIAL HEAD LCCATION TC MIDPT
      INITIAL     X3,20000        SET INITIAL HEAD LCCATION TO MIDPT
      INITIAL     X4,20000        SET INITIAL HEAD LCCATICN TO MICPT
      INITIAL     X5,20000        SET INITIAL HEAD LCCATION TC MIDPT
      INITIAL     X6,0            SET INITIAL WRITE TAPE HEAD TO LOCATICN 0
      INITIAL     X73,150         MAX # TBM BLKS ON DISC BEFCRE WRITE
      INITIAL     X75,0           FOR SAVINGS COUNTER
      INITIAL     X76,1           MINIMUM # CF JOBS BEFCRE PROCESSING
      INITIAL     X77,6           WRITE TAPE IS THE LAST TAPE

1     STORAGE     1               DATA CHANNELS AVAILABLE IN SYSTEM
2     STORAGE     1               1 DISK CCNTROLLER AVALIABLE
3     STCRAGE     1               1 SUPERVISOR CONTRCL PROCESSOR
4     STORAGE     6400            MAX # BLCCKS ON 8 3330 DISCS

1     VARIABLE    P2*K10          HOST PROCESSING TIME - LINEAR
2     FVARIABLE   10*(24/10+(P1-K1230)/1000) SEARCH TIME > 1230 BLOCKS
3     FVARIABLE   9/2+P2*(19/10)             READ TIME AT 5 IPS
4     VARIABLE    P6-P1           COMPUTES CISTANCE TC BE SEARCHED
5     VARIABLE    P1*(KO-K1)      MAKE SEARCH DISTANCE POSITIVE
6     VARIABLE    K3*V3           WRITE TIME (WRITE, VERIFY, TALLY)
7     VARIABLE    P6+P2           COMPUTE NEW HEAD LOCATION AFTER READ
8     VARIABLE    X74-K1          CECREMENT # OF TAFE DRIVES AVALIABLE
9     VARIABLE    X74+K1          INCREMENT # OF TAPE DRIVES AVALIABLE
10    VARIABLE    X75+1           FOR A CCUNTER
11    VARIABLE    X*1-K1          TO DECREMENT FINISHED REQUESTS CCUNTER

    1 TABLE       P3,1,1,7        TO GATHER PRIORITY STATISTICS
    2 TABLE       P2,0,3,3        TO GATHER DATA SET SIZE STATISTICS
    3 TABLE       P4,1,1,9        TO GATHER DATA SETS / JOB STATISTICS
    4 TABLE       P5,1,1,6        TOPPLATE USAGE STATISTICS
    5 TABLE       P6,1000,1000,42 TO GATHER DATA SET LOCATION STATISTICS
    6 QTABLE      38,0,20,25          WAIT FOR TRANSPCRT DRIVE STATISTICS
    7 QTABLE      37,0,100,100        TIME IN SYSTEM STATISTICS

      INITIAL     X74,4           NUMBER OF TRANSPORT CRIVES
      START       100,NP          TO INITIALIZE THE SYSTEM
      RESET                       ELIMINATE INITIAL STATISTICS
      START       1000            GATHER STATISTICS FCR X #.OF DS REQUESTS
      CLEAR       X1-X74,X76-X77    CLEAR MODEL FCR ANOTHER RUN

      INITIAL     X74,3           NUMBER CF TRANSPORT CRIVES
      START       100,NP          TO INITIALIZE THE SYSTEM
```

```
RESET                      ELIMINATE INITIAL STATISTICS
START       1000           GATHER STATISTICS FOR X # OF DS REQUESTS
CLEAR       X1-X74,X76-X77   CLEAR MODEL FCR ANOTHER RUN

INITIAL     X74,2          NUMBER OF TRANSPORT DRIVES
START       100,NP         TO INITIALIZE THE SYSTEM
RESET                      ELIMINATE INITIAL STATISTICS
START       1000           GATHER STATISTICS FCR X # OF DS REQUESTS
CLEAR       X1-X74,X76-X77   CLEAR MODEL FCR ANOTHER RUN

INITIAL     X74,1          NUMBER CF TRANSPORT DRIVES
START       100,NP         TO INITIALIZE THE SYSTEM
RESET                      ELIMINATE INITIAL STATISTICS
START       1000           GATHER STATISTICS FCR X # OF DS REQUESTS
END
```

BIBLIOGRAPHY

1. Ampex Corporation: Terabit Memory Systems, A Brief Description of the Terabit Memory System, 7 April 1971.

2. Ampex Corporation: TBM Systems Department Report B6-972, The TBM* Memory System.

3. Naval Postgraduate School, A Survey and Analysis of High Density Mass Storage Devices and Systems, by N. F. Schneidewind, G. H. Syms, T. L. Grainger, and R. J. Carden, July 1972.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0212 Naval Postgraduate School Monterey, California 93940	2
3.	Naval Postgraduate School Department of Operations Research and Administrative Sciences Monterey, California 93940	1
4.	Chief of Naval Personnel Attn: Pers 11b Department of the Navy Washington, D. C. 20370	1
5.	Assistant Professor G. H. Syms, Code 53ZZ Department of Mathematics Naval Postgraduate School Monterey, California 93940	1
6.	Professor N. F. Scheidewind, Code 55SS Department of Operations Research and Administrative Sciences Naval Postgraduate School Monterey, California 93940	1
7.	ENS. R. B. Frazier, USN 312 North Luna Court Hollywood, Florida 33021	1
8.	Professor L. J. Prince Systems Analysis Department Miami University Oxford, Ohio 45056	1
9.	TBM* Memory System Department Ampex Corporation 1020 Knifer Road Sunnyvale, California 94086	1

10. O.I.C. TBM* Memory System 1
Electronic Data Processing Department
Fort George G. Meade, Maryland 20755

11. Mr. John Fekula 1
Information Service Center
Central National Bank
120 South LaSalle
Chicago, Illinois 60603

12. Chairman, Computer Science Group, Code 72 BV 1
Naval Postgraduate School
Monterey, California 93940

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)* Naval Postgraduate School Monterey, California 93940	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE

Simulation Analysis of the TBM* Memory System: An On-Line Mass Storage System

4. DESCRIPTIVE NOTES *(Type of report and, inclusive dates)*

Master's Thesis; June 1973

5. AUTHOR(S) *(First name, middle initial, last name)*

Robert B. Frazier

6. REPORT DATE June 1973	7a. TOTAL NO. OF PAGES 141	7b. NO. OF REFS 3

8a. CONTRACT OR GRANT NO. b. PROJECT NO. c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) 9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*

10. DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Naval Postgraduate School Monterey, California 93940

13. ABSTRACT

The TBM* Memory System is an on-line, high density memory system with a capacity of up to three trillion bits. It is a cost efficient method of storing large data bases on-line with an effective response similar to disc storage.

Some features of the TBM* Memory System are: 1) Complete system redundance; 2) Modular expansion; 3) Hardware and data transfer reliability; 4) Large data base on-line; 5) Rapid response; and 6) Low cost per bit.

Major applications of a mass storage system are: 1) The replacement of large conventional on-line and off-line tape library systems; 2) Supplements to enlarge disc memory systems; and 3) the provision of on-line storage for large data bases.

Since the TBM* Memory System is indeed very complicated, a simulation model was developed to help the user understand the system and to predict the TBM* performance under various configurations and loading conditions. In order to help the user select the proper configuration for his installation, a sensitivity analysis is also provided.

KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
MASS STORAGE SYSTEM						
TBM* MEMORY SYSTEM						
SUPERVISORY CONTROL PROCESSER						
EXTERNAL DATA CHANNEL PROCESSER						
DATA CHANNEL						
TRANSPORT DRIVER						
DUAL TRANSPORT MODULE						
TOPPLATE						
MEGABYTE						
HOST INSTALLATION						
TRILLION BITS						

23833
21 MAY 76

Thesis 144059
F7867 Frazier
c.1 Simulation analysis of
the TBM* memory system:
An on-line mass storage
system.